u évora
ESCOLA DE CIÊNCIAS E TECNOLOGIA

JIUE '2011
2as Jornadas de Informática da Universidade de Évora

Actas das
2as Jornadas de Informática da Universidade de Évora
JIUE'2011

Luís Rato e Teresa Gonçalves

Évora, Portugal
16 de Novembro de 2011

Escola de Ciências e Tecnologia
Universidade de Évora
2011

ISBN: 978-989-97060-2-6

http://host.di.uevora.pt/jiue2011

Sobre o evento JIUE'2011

No dia 16 de Novembro de 2011, as 2ªs Jornadas de Informática da Universidade de Évora (JIUE'2011) juntaram docentes de vários departamentos, alunos de licenciatura, mestrado e doutoramento. Este evento tem como objetivo divulgar os trabalhos desenvolvidos na área de Informática, no âmbito de teses de mestrado e doutoramento, projetos de investigação e colaboração com empresas realizados em ligação com Unidades Orgânicas da Universidade de Évora.

Dando continuidade ao JIUE'2010[1], o Departamento de Informática organizou a segunda edição deste evento, que congrega diversos sectores da Instituição, tendo como pano de fundo a investigação ou o desenvolvimento de serviços inovadores na área de Informática.

Comparativamente ao ano anterior[2], registou-se mais uma submissão (19 artigos). Após o processo de revisão dupla pelos membros da Comissão de Programa[3], foram aceites 17 trabalhos para apresentação oral no evento. A maioria dos trabalhos é relacionada com alunos de 2º e 3º ciclo sob orientação de docentes do Departamento de Informática. Houve uma submissão do Departamento de Engenharia Rural. Entre os oradores desta edição das Jornadas, destaca-se a participação de alunos visitantes de nacionalidade estrangeira, especificamente do Brasil e do Nepal.

A organização do evento agradece a todos os participantes.

[1] http://host.di.uevora.pt/jiue2010
[2] http://host.di.uevora.pt/docs/actas-jiue2010_final.pdf
[3] http://host.di.uevora.pt/jiue2011/organizacao

Comissão Organizadora

- José Saias
- Iara Almeida
- Pedro Patinho

Comissão de Programa

- Carlos Caldeira
- Francisco Coelho
- Iara Almeida
- Irene Rodrigues
- José Saias
- Lígia Ferreira
- Luis Rato
- Paulo Quaresma
- Teresa Gonçalves
- Vitor Nogueira

JIUE'2011

Sessão 1

Sessão 2

Sessão 3

Cooperative Question Answering for the Semantic Web

Dora Melo[1], Irene Pimenta Rodrigues[2], and Vitor Beires Nogueira[2]

[1] Instituto Politécnico de Coimbra and CENTRIA, Portugal
dmelo@iscac.pt,
[2] Universidade de Évora and CENTRIA, Portugal
{ipr,vbn}@di.uevora.pt

Abstract. In this paper we propose a Cooperative Question Answering System that takes as input queries expressed in natural language and is able to return a cooperative answer obtained from resources in the semantic web, more specifically DBpedia represented in OWL/RDF as knowledge base and WordNet to build similar questions. Our system resorts to ontologies not only for reasoning but also to find answers and is independent of prior knowledge of the semantic resources by the user. The natural language question is translated into its semantic representation and then answered by consulting the semantics information sources. If there are multiple answers to the question posed (or to the similar questions for which DBpedia contains answers), they will be grouped according to their semantic meaning, providing a more cooperative and clean answer to the user.

Keywords: Natural Language, Ontology, Question Answering, Semantic Web

1 Introduction

Ontologies and the semantic web [1] became a fundamental methodology to represent the conceptual domains of knowledge and to promote the capabilities of semantic question answering systems [2]. By allowing search in the structured large databases and knowledge bases of the semantic web these systems can be considered as an alternative or as a complement to the current web search.

There is a gap between users and the semantic web: it is difficult for end-users to understand the complexity of the logic-based semantic web. Therefore it is crucial to allow a common web user to profit from the expressive power of semantic web data models while hiding its potential complexity. There is a need for user-friendly interfaces that scale up to the web of data and support end-users in querying this heterogeneous information source.

In this paper we propose a cooperative question answering system that is independent of prior knowledge of the semantic resources by the user and is able to answer cooperatively to questions posed in natural language. This system maintains the structure of the dialogue that provides a context for the interpretation of the questions and includes implicit content such as spatial and temporal

knowledge, entities and information useful for the pragmatic interpretation like discourse entities used for anaphora resolution. The system starts a dialogue whenever there is some question ambiguity or when it detects that the answer is not what user expected. Our proposal considers only the English natural language and includes deep parsing, use of ontologies, lexical and semantic resources such as the WordNet [3] and web resources like the DBpedia [4].

This paper is organized as follows. First, in Section 2, we introduce the proposed system, describing the main components of its architecture. In parallel, we present an example as an illustration of the system functionality. Afterwards, in Section 3 we present related work, highlighting the main differences in the proposed system. Finally, in Section 4, we present the conclusions and the future work.

2 Proposed System

Very briefly, the proposed system receives a natural language question and translates into a semantic representation using Discourse Representation Structures (DRS). Then, after consulting the semantics sources of information, provides a natural language answer. If there are multiple answers to the question posed (or to the similar questions for which DBpedia contains answers), they will be grouped according to their semantic meaning, providing a more cooperative and clean answer to the user. Therefore, we consider that our system provides a user friendly interface.

The language chosen for our system was Prolog with several extensions and libraries. Among the reasons for such choice is the fact that there is a wide range of libraries for querying and processing of ontologies OWL2, WordNet has an export for Prolog and there are extensions that allow us to incorporate the notion of context into the reasoning process.

Our system architecture is presented in Figure 1 and to help its understanding we describe the main components in the following subsections.

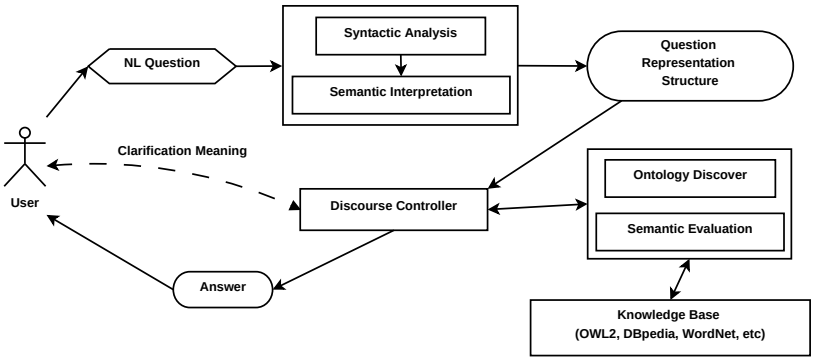


Fig. 1. Question Answering System Architecture.

2.1 Semantic Interpretation

Semantic analysis (or interpretation) is built using first-order logic [5] extended with generalized quantifiers [6]. We take special care with the discourse entities in order to have the appropriate quantifier introduced by the determinant interpretation. At this step, the syntactic structure of the question is rewritten into a DRS[3], that is supported by Discourse Representation Theory [7].

As an illustration, consider the question "All French romantic writers have died?". The syntactic analysis generates a tree that is rewritten according to a set of rules and integrated into a DRS. In our study, it is stated by the following representation structure:

```
drs([all-X, exist-Y],[writer(Y), french(Y), romantic(Y)],
                                                [died(X)])
```

where the referent of the discourse is `all-X`, with `X` an universally quantified discourse entity, the main predication of the question is `died(X)` and the presupposed predications are `writer(Y), french(Y), romantic(Y)`, with `Y` an existential quantified discourse entity. The system has to find all entities `X` that must verify the main predication condition only for those entities `Y` that verify all the question presupposed conditions. The answer to the question will be a list with all french, romantic, writers resource entities who died.

2.2 Ontology Discovery

The Ontology Discovery is guided by the Discourse Controller to obtain the extension of sentence representation along with the reasoning process. The reasoning context and the question meaning will change whenever the Discourse Controller reaches a dead end.

This system module looks for similarities between labels according to their string-based, taking into account abbreviations, acronyms, domain and lexical knowledge. If an answer is not achieved, each term in the query is extended with its synonyms, hypernyms and hyponyms obtained from WordNet[4] [8]. Afterwards we extract a set of semantic resources which may contain the information requested.

Continuing the example of the previous section, in order to obtain the extension of sentence representation along the reasoning process, the system has to find the classes, properties or instances that have labels matching the search terms 'writer', 'french', 'romantic' and 'died', either exactly or partially. For instance, to represent the concept 'writer', the system finds the DBpedia class `Writer` [5], with property domain `Work` and domain range `Person`.

[3] For us a DRS is a set of referents, universally quantified variables and a set of conditions (first-order predicates). The conditions are either atomic (of the type $P(u_1, ..., u_n)$ or $u_1 = u_2$) or complex (negation, implication, disjunction, conjunction or generalized quantifiers).

[4] `http://wordnet.princeton.edu/`

[5] `http://dbpedia.org/ontology/Writer`

If the system did not find any correspondence to a word and its derivatives, the user is informed and can clarify the system by reformulating the question or presenting others query(ies).

2.3 Semantic Evaluation

Semantic evaluation is intended to be the pragmatic evaluation step of the system, where the question semantic is transformed into a constraint satisfaction problem. This is achieved by adding conditions that constrain the discourse entities. Moreover, this extra information (regarding the question interpretation) can help the Discourse Controller to formulate a more objective answer.

The semantic evaluation must reinterpret the semantic representation of the sentence, based on the ontology, in order to obtain the set of facts that represent the information provided by the question.

Back to our example, to solve the constraint problem the Dialogue Controller generates and poses the questions such "Who are the French romantic writers?" to the question answering system, whose representation structure is

```
drs([wh-X,exist-Y],[writer(Y), french(Y), romantic(Y),
                                                is(X,Y)]).
```

First and according to the domain knowledge, the interpreter will transform the conditions of the DRS into OWL. For instance, the condition `ontology_writer` will represent the DRS condition `writer`. Therefore, the new representation structure[6] for the question is

```
drs([wh-X,exist-Y],[ontology_writer(Y), ontology_french(Y),
                              ontology_romantic(Y), is(X,Y)])
```

After obtaining this new set of DRS, the terms of the ontology will be interpreted as usual Prolog predicates. Then, by applying the unification mechanism of Prolog the system will obtain the answer to the question. Therefore, the answer to initial question will be

```
Francois-Rene de Chateaubriand (1768-1848)
Alphonse de Lamartine (1790-1869)
Alfred de Musset (1810-1857)
Victor Hugo (1802-1885)
Henri-Marie Beyle, Stendhal (1783-1842)
```

2.4 Discourse Controller

The Discourse Controller is a core component that is invoked after the natural language question has been transformed into its semantic representation. Essentially the Discourse Controller tries to make sense of the input query by looking

[6] The condition `ontology_term` represents the class, property or instance in the ontology that is the meaning of the term. If the interpreter has more than one possible ontology conditions for each term then will get several DRS rewritten with the terms of the ontology.

at the structure of the ontology and the information available on the semantic web, as well as using string similarity matching and generic lexical resources.

The Dialogue Controller deals with the set of discourse entities and is able to compute the question answer. It has to verify the question presupposition, choose the sources of knowledge to be used and decide when the answer has been achieved or to iterate using new sources of knowledge. The decision of when to relax a question in order to justify the answer and when to clarify a question and how to clarify it also taken by in this module.

Whenever the Discourse Controller isn't sure how to disambiguate between two or more possible terms or relations in order to interpret a query, it starts a dialogue with the user and asks him for disambiguation. The clarification done by the user will be essential for the Discourse Controller, this way obtaining the right answer to the query posed by the user. If there are multiple answers to the question posed by the user (or to the similar questions for which DBpedia contains answers), they will be grouped according to their semantic meaning, providing a more cooperative and clean answer to the user. To do so, the Discourse Controller has to reason over the question and construct the answer.

Our dialoguing system has as main objective the use of interaction to obtain more objective and concrete answers. It is not used only to clarify the problems of ambiguity, but also to help finding the path to the correct answer. Making the dialogue system more cooperative makes one able to get closer to the answer desired by the user. In many cases, the user is the only one who can help the system in the deduction and interpretation of information.

3 Related Work

The representation of questions with generalized quantifiers as in [9] allows the use of various natural language quantifiers like all, at least 3, none, etc. Moreover, the question evaluation also resorts to logic programming with constraints.

A query language for OWL based on Prolog is presented in [10]. The author proposes a way of defining a query language based on a fragment of Description Logic and a way of mapping it into Prolog by means of logic rules.

In [11] we find a declarative approach to represent and reason about temporal contextual information. In this proposal each question takes place in a temporal context and that context is used to restrict the answer.

PowerAqua [12] is a multi-ontology-based question answering system that takes as input queries expressed in natural language and is able to return answers drawn from relevant distributed resources on the semantic web.

Our proposal is a friendly, simple and cooperative question answering system. The main difference is the cooperative way on answering the natural language questions posed by the user. We interact with the user in order to disambiguate and/or to guide the path to obtain the correct answer to the query, whenever this is possible to do by the reasoner. We also use cooperation to provide more informed answers. The answers have to clarify what the system can infer about the question from the knowledge domain.

4 Conclusions and Future Work

We presented a cooperative semantic web question answering system that receives queries expressed in natural language and is able to return a cooperative answer, also in natural language, obtained from semantic web resources. The system is able of dialoguing when the question has some ambiguity or when it detects that the answer is not what the user expected. Our proposal includes deep parsing, the use of ontologies, lexical and semantic resources such as the WordNet and web resources like the DBpedia.

As future work, we intend to answer more complex questions and extend it to Portuguese natural language. For this purpose, it will be necessary to enrich the knowledge domain with concepts that may be deduced from the initial domain. Although the system is intended to be domain independent, it will be tested in a number of domains, with special relevance to the wine and the cinema, since for these fields there are many resources available in the semantic web. We contemplate about enlarging the knowledge base with other ontologies in order to support open domain question answering and take advantage of the vast amount of heterogeneous semantic data provided by the semantic web.

References

1. Horrocks, I.: Ontologies and the semantic web. Communications of the ACM **51** (2008) 58
2. Guo, Q., Zhang, M.: Question answering based on pervasive agent ontology and Semantic Web. Knowledge-Based Systems **22** (2009) 443–448
3. Fellbaum, C.: WordNet: An electronic lexical database. The MIT press (1998)
4. Auer, S., Bizer, C., Kobilarov, G., Lehmann, J.: Dbpedia: A nucleus for a web of open data. The Semantic Web **4825** (2007) 722–735
5. Hodges, W.: Classical logic I: first-order logic. The Blackwell guide to philosophical logic (2001) 9–32
6. Barwise, J., Cooper, R.: Generalized quantifiers and natural language. Linguistics and philosophy **4** (1981) 159–219
7. Kamp, H., Reyle, U.: From Discourse to Logic. Volume 42 of Studies in Linguistics and Philosophy. Kluwer (1993)
8. Witzig, S., Center, A.: Accessing wordnet from prolog. Artificial Intelligence Centre, University of Georgia (2003) 1–18
9. Rodrigues, I., Quintano, L., Ferreira, L.: Nl database dialogue question-answering as a constraint satisfaction problem. In: 18th Intl. Conf. on Applications of Declarative Programming and Knowledge Management (INAP'09), Univ. Évora (2009)
10. Almendros-Jiménez, J.M.: A Prolog-based Query Language for OWL. Electronic Notes in Theoretical Computer Science **271** (2011) 3–22
11. Nogueira, V., Abreu, S.: Temporal contextual logic programming. Electronic Notes in Theoretical Computer Science **177** (2007) 219–233
12. Lopez, V., Motta, E.: Poweraqua: Fishing the semantic web. Semantic Web: Research and Applications (2006)

Classificação de Argumentos Sintácticos Aproximação preliminar

João Sequeira, Teresa Gonçalves, and Paulo Quaresma

Universidade de Évora
m5071@alunos.uevora.pt, tcg@uevora.pt, pq@uevora.pt

Resumo Este artigo apresenta uma aproximação preliminar de uma vertente pouco explorada do processamento de linguagem natural para a língua Portuguesa, a classificação de argumentos sintácticos.
Primeiro é dada uma introdução à classificação de argumentos sintácticos, posteriormente são explicados os passos necessários à criação de um classificador utilizando a ferramenta MinorThird. O desempenho foi verificado nos argumentos sintácticos mais frequentes (predicado, sujeito e complemento directo) num subconjunto do Bosque 8.0. A mesma abordagem foi aplicada a um corpus da língua Inglesa utilizado no CONLL 2004 e os resultados foram comparados com os obtidos na tarefa conjunta do CONLL 2004.

1 Introdução

Actualmente existe uma grande quantidade de conteúdos digitais de cariz académico, pessoal, noticioso entre outros disponíveis para consulta na Internet. A tarefa de obter informação de conteúdos não tratados de fontes tão dispares tornou-se praticamente impossível [14,15].
Com o incremento de conteúdos digitais publicados existiu também um aumento na pesquisa de aplicações que consigam analisar e extrair informação automaticamente dos mesmos. Este factor tem proporcionando nos últimos anos uma crescente procura de aplicações de processamento de linguagem natural[1] [4].
A classificação de argumentos sintácticos[2] tem vindo a ser uma área de cada vez mais interesse, devido à sua crescente importância em sistemas de extracção de informação, pergunta-resposta, sumarização de documentos entre outras aplicações que necessitam de informação semântica [3]. Esta vertente do processamento de linguagem natural já possui vários recursos disponíveis para línguas como o Inglês, produto de vários projectos apresentados ou implementados para conferências internacionais [3]. Mas ainda existe muita matéria a ser explorada no âmbito de outras línguas, estando o Português entre elas.
Este trabalho explora a utilização da ferramenta MinorThird[3] [5] na tarefa de

[1] Do Inglês *Natural Language Processing (NLP)*, sendo possível para o português usar também a sigla PLN
[2] Nas conferências internacionais normalmente é usado o termo *semantic role labelling*, retratando as relações semânticas entre os diferentes constituintes duma frase
[3] http://sourceforge.net/apps/trac/minorthird/wiki

classificação de argumentos sintácticos para a língua Portuguesa. Para possuir um meio de comparação com os resultados obtidos internacionalmente é feita a mesma tarefa com o corpus em Inglês usado no CONLL[4] 2004 [3].

2 Classificação de Argumentos Sintácticos

A classificação de argumentos sintácticos é actualmente um dos subgrupos mais activos na área de processamento de linguagem natural. Nos últimos 10 anos o grupo SIGNLL[5] e todos os sistemas participantes abordaram este tema nas edições 2004 [3], 2005 [4], 2008 [22] e 2009 [9] das conferencias CONLL.
Consiste em identificar os verbos presentes numa frase e os seus argumentos sintácticos [4], tais como sujeito da acção, objecto da acção entre outros.

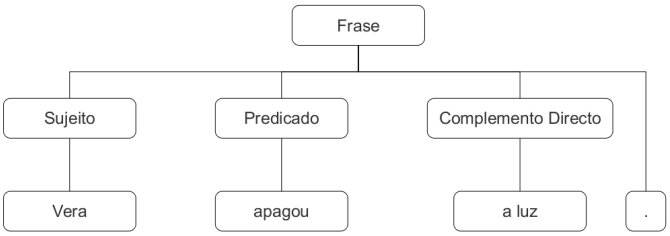


Figura 1. Frase do Bosque classificada com os respectivos argumentos sintácticos.

É visível na Figura 1 a classificação de uma frase presente no corpus usado para a língua Portuguesa, Bosque[6] [1]. Ao analisar a frase verifica-se que possui um sujeito ("Vera"), um verbo que compõe o predicado ("apagou") e um complemento directo que sofre a acção realizada pelo sujeito ("a luz").
Gildea e Jurafsky, pioneiros na classificação de argumentos sintácticos, enumeram dois métodos proeminentes para realizar a análise de textos, um baseado em gramáticas[7] e outro orientado a dados. O processo de criar gramáticas é muito moroso visto serem criadas à mão e necessitarem de incluir uma descrição para cada caso existente na língua[8]. Para os sistemas orientados a dados são necessários corpora classificados, e aplicações capazes de criar modelos a partir dos mesmos. Esses modelos são posteriormente usados para classificar textos sem marcações. Exemplos dessas aplicações são os participantes nas tarefas conjuntas[8] das edições enumeradas anteriormente das conferências CONLL [3,4].

[4] *Conference on Computational Natural Language Learning*, `http://ifarm.nl/signll/conll/`
[5] Special Interest Group on Natural Language Learning
[6] http://www.linguateca.pt/Floresta/corpus.html
[7] Do Inglês *grammar based systems*
[8] Do Inglês *Shared Task*

3 Classificador de Argumentos Sintácticos

3.1 Pré-Processamento dos Corpora

O corpus usado para criar o modelo baseia-se no Bosque 8.0[9] incorporado no projecto Floresta Sintá(c)tica[1]. A Floresta Sintá(c)tica consiste em texto corrido dividido em frases analisadas sintacticamente em estruturas de árvore pelo analisador sintáctico PALAVRAS [2] [1]. O Bosque 8.0 é composto por 9368 frases dos primeiros 1000 extractos do CETEMPúblico e do CETEMFolha, priorizando a qualidade em detrimento da quantidade [12]. Para o CETEMPúblico foram usados excertos de noticias retiradas do jornal Público [20] e para o CETEMFolha foram usados excertos de noticias retiradas do jornal Folha de S. Paulo [6].

```
'source' => 'CP429-7 Vera apagou a luz.',
'number' => 1,
'cod' => 'CETEMPúblico n=429 sec=clt sem=96a',
't' => [
  'fcl||STA',
  [
    'np||SUBJ',
    'prop(\'Vera\' F S)||H::Vera'
  ],
  [
    'vp||P',
    'v-fin(\'apagar\' PS 3S IND)||MV::apagou'
  ],
  [
    'np||ACC',
    'art(\'o\' <artd> F S)||>N::a',
    'n(\'luz\' <np-def> F S)||H::luz'
  ],
  'jjpunct(-.-)'
]
```

Figura 2. Representação da frase '`Vera apagou a luz.`' presente no Bosque.

Neste trabalho foi usado o CETEMPúblico, tendo sido removidas as frases consideradas títulos de noticias. O pré-processamento do corpus, no final do mesmo com 4416 frases, foi dividido nos seguintes passos:

1. extraiu-se as palavras e respectivas categorias sintácticas das frases. As frases estavam dispostas na forma visível na Figura 2;

[9] Obtido em: `http://www.linguateca.pt/Floresta/corpus.html`

2. converteu-se as categorias sintácticas em etiquetas XML[10], formando um conjunto de frases com a forma visível na Figura 3.

```
<SUBJ>Vera<\SUBJ> <P>apagou<\P> <ACC>a luz<\ACC>.
```

Figura 3. Representação da frase '`Vera apagou a luz.`' com etiquetas XML.

Um processamento similar foi realizado com o corpus utilizado na tarefa conjunta do CONLL de 2004[11] [3]. Foi seleccionado o corpus do CONLL 2004 visto ter sido a primeira abordagem à tarefa de classificação de argumentos sintácticos realizada nestas conferências. Como este artigo documenta uma primeira aproximação à mesma tarefa mas para o Português achou-se por bem tentar estar no mesmo patamar para obter uma melhor medição do desempenho do classificador. Este corpus foi composto por seis secções do jornal de Wall Street [7] (15 a 18 para treino (8936 frases), 20 para desenvolvimento (1671 frases) e 21 para teste (2012 frases)) presentes no Penn Treebank [17,13] ao qual foi acrescentado a informação de estruturas sintácticas de predicado-argumento presentes no PropBank [10,16].

3.2 MinorThird

O MinorThird é um conjunto, em código aberto, de classes implementadas na linguagem de programação Java para realizar tratamento de textos como por exemplo guardar, anotar e categorizar textos, aprendizagem e extracção de entidades mencionadas. Foi criado pelo professor William W. Cohen da Universidade de Carnegie Mellon e actualmente é mantido Frank Lin [5].
O MinorThird usa colecções de documentos para criar uma base de dados denominada *TextBase* sobre a qual são realizadas afirmações lógicas para posteriormente serem guardadas num objecto do tipo *TextLabels*. Como a anotação presente no objecto *TextLabels* é independente do conteúdo dos documentos podem existir vários tipos de anotações para o mesmo conjunto de documentos [5]. As anotações no *TextLabels* enumeram as categorias ou propriedades, podendo ser sintácticas ou semânticas, de uma palavra, documento ou *span*[12]. Estas anotações podem ser criadas manualmente, ou automaticamente através de uma aplicação. Os *TextLabels* e as *TextBases* a eles associadas podem ser guardados num repositório previamente configurado [5].
Os métodos de aprendizagem de extracção e classificação de *spans*, subconjuntos de palavras de um documento ou documentos inteiros, presentes no MinorThird são numerosos. Entre os métodos de aprendizagem sequencial estado-da-arte estão os campos condicionais aleatórios[13] [24,11] e métodos de treino de mo-

[10] Sigla do Inglês *Extensible Markup Language*
[11] Obtido em: `http://www.lsi.upc.edu/~srlconll/st04/st04.html`
[12] Em português enumera um conjunto de palavras
[13] Do Inglês *Conditional Random Fields*

delos escondidos de Markov[14] [21] [5]. Sendo o MinorThird uma ferramenta de aprendizagem supervisionada orientada a dados os passos do seu funcionamento estão representados na Figura 4:

- num primeiro passo é criado um modelo (*TextLabels*) utilizando ficheiros classificados (*TextBases*);
- num segundo passo esse modelo é usado para classificar textos sem marcações dando como resultado os textos classificados.

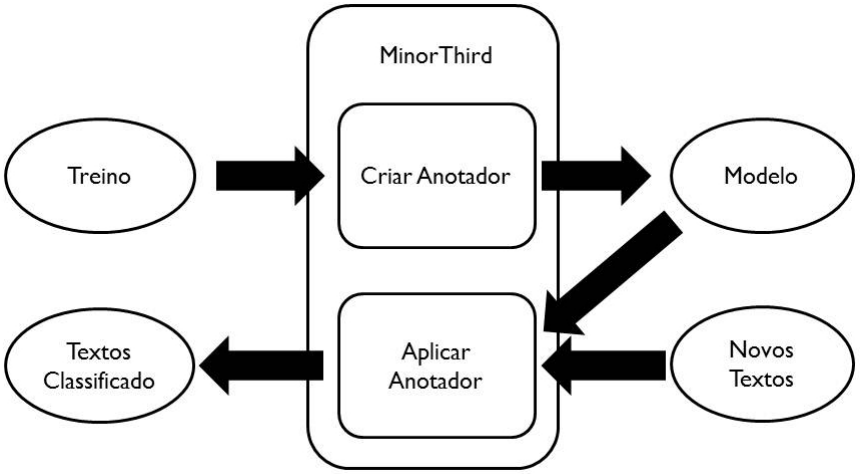


Figura 4. Funcionamento do MinorThird, criando um modelo com base em ficheiros de treino e posteriormente usando esse modelo para classificar novos textos.

3.3 Cenário Experimental

Os corpora obtidos no pré-processamento, e renomeados para uma melhor compreensão na análise dos resultados:

- BosqueXML: corpus criado com base no Bosque 8.0. Das classes presentes no Bosque 8.0 foram utilizadas no BosqueXML apenas as que forneciam viabilidade estatística para a tarefa da classificação de argumentos sintácticos.
- CONLL'2004: corpus obtido após processamento do usado na tarefa conjunta do CONLL 2004.

A Tabela 1 apresenta o número de *spans* para cada argumento sintáctico estudado em cada corpora.

[14] Do Inglês *Hidden Markov Models*

		BosqueXML	CONLL'2004	
Etiqueta	Descrição	#	# Treino	#Teste
`P`	Predicado	7268	19098	3627
`Arg0`	Sujeito	4673	12709	1671
`Arg1`	Complemento Directo	3802	18046	3429

Tabela 1. Etiquetas, descrição e contagens para os corpora BosqueXML e CONLL'2004.

Foram testados os algoritmos existentes no MinorThird sendo que os melhores resultados foram obtidos com os algoritmos SVMCMM e CRF. O algoritmo SVMCMM é constituido por modelos condicionais de Markov[15] [19,18] treinados com máquinas de vectores de suporte[16] [23]. O algoritmo CRF utiliza campos condicionais aleatórios [24,11].
Todos os testes feitos usaram as características pré-definidas de cada algoritmo com uma janela de contexto de tamanho três.

Foi aplicado o procedimento validação cruzada 10 pastas ao BosqueXML e um procedimento treino/teste ao CONN'2004. O desempenho dos modelos foi analisado com recurso à precisão (π), cobertura (ρ) e medida F_1 obtidas na classificação dos *spans*.

3.4 Resultados

A Tabela 2 mostra os resultados obtidos com os algoritmos SVMCMM e o CRF utilizando o corpus BosqueXML. Observa-se que o algoritmo CRF apresenta melhores valores de precisão enquanto que o SVMCMM apresenta melhores valores de cobertura.

Para ambos os algoritmos verifica-se que os valores da precisão estão pelo menos 0.1 acima dos da cobertura para todas as etiquetas (para o algoritmo CRF o `Arg0` e `Arg1` apresentam valores de precisão superiores em 0.2 quando comparados com a cobertura). A etiqueta *Predicado* apresenta os melhores resultados com valores de F_1 acima dos 54% enquanto que o *Complemento Directo* apresenta valores de F_1 abaixo dos 21%.

A Tabela 3 mostra os resultados obtidos com os algoritmos SVMCMM e CRF utilizando o corpus CONLL'2004.

O corpus CONLL'2004 apresenta valores de precisão e cobertura similares para ambos os algoritmos (excepto para a etiqueta `Arg0` onde o CRF tem um valor 0.1 superior na precisão). Mais uma vez o *Predicado* apresenta melhores resultados com valores de F_1 acima dos 82%, enquanto que o *Complemento Directo* tem valores de F_1 abaixo dos 24%.

Comparando a Tabela 2 e a Tabela 3 pode-se concluir que os resultados obtidos com o corpus da língua Portuguesa estão abaixo dos obtidos com o

[15] Do Inglês *Conditional Markov Models*
[16] Do Inglês *Support Vector Machines*

	SVMCMM			CRF		
Etiqueta	π	ρ	F_1	π	ρ	F_1
`P`	.603	.503	.548	.660	.475	.545
`Arg0`	.416	.283	.337	.447	.237	.308
`Arg1`	.285	.161	.206	.361	.117	.175

Tabela 2. Precisão, cobertura e F_1 do corpus BosqueXML utilizando os algoritmos SVMCMM e CRF.

	SVMCMM			CRF		
Etiqueta	π	ρ	F_1	π	ρ	F_1
`P`	0.850	0.823	0.836	0.842	0.805	0.823
`Arg0`	0.599	0.464	0.523	0.699	0.463	0.557
`Arg1`	0.372	0.170	0.234	0.414	0.151	0.221

Tabela 3. Precisão, cobertura e F_1 do corpus CONLL'2004 utilizando os algoritmo SVMCMM e CRF.

corpus da língua Inglesa. Uma possível explicação para esta diferença poderá ser o tamanho dos corpora: o CONLL'2004 é sensivelmente 3 vezes maior que o BosqueXML. Outra explicação possível é a estrutura sintáctica da língua Inglesa ser mais simples que a da língua Portuguesa.

A Tabela 4 compara os melhores valores de F_1 para as etiquetas `Arg0` and `Arg1` obtidos com o MinorThird (`Arg0` com o algoritmo CRF e `Arg1` com o algoritmo SVMCMM) com o melhor e pior obtidos na tarefa conjunta do CONLL 2004 como reportado em [3] (os valores dos *Predicados* não são mostrados visto estes não terem sido considerados na avaliação da tarefa conjunta).

		CONLL 2004	
Etiqueta	MinorThird	Melhor	Pior
`Arg0`	0.557	0.814	0.562
`Arg1`	0.234	0.716	0.490

Tabela 4. Máximo, mínimo dos valores de F_1 dos argumentos mais comuns da tarefa conjunta do CONLL 2004 comparados com os valores obtidos usando o corpus CONLL'2004 com o MinorThird.

Da Tabela 4 pode-se observar que o uso de informação linguística adicional como por exemplo categorias gramaticais, sintagmas, orações e entidades mencionadas é proveitosa para a tarefa de classificação de argumentos sintácticos enquanto que métodos sequenciais de classificação não são suficientes. O uso desta informação melhora o desempenho dos sistemas como é visível quando se compara os valores de `Arg0` e `Arg1` obtidos com o MinorThird e dos sistemas do CONLL 2004. A diferença é superior no *Complemento Directo* do que no *Sujeito*.

4 Conclusões e Trabalho Futuro

Este artigo tentou realizar uma pesquisa para a língua Portuguesa numa área ainda pouco explorada. Verificou-se que os resultados obtidos com um corpus da língua Portuguesa estão abaixo dos obtidos com um corpus da língua Inglesa. Como já foi mencionado anteriormente a diferença de desempenho pode dever-se com o diferente tamanho dos corpora. Outra possível explicação é o uso de estruturas sintácticas mais complexas e as muitas contracções de palavras existentes na língua Portuguesa quando comparada com a língua Inglesa.

Verificou-se que informação linguística tais como palavras, categorias gramaticais, sintagmas, orações e entidades mencionadas são úteis para a tarefa de classificação de argumentos sintácticos e o uso de apenas métodos de aprendizagem sequenciais não produzem bons resultados.

Como trabalho futuro temos a intenção de aumentar o tamanho do corpus da língua Portuguesa e desenvolver um classificador que faça uso de toda a informação linguística mencionada anteriormente. Apenas nesses termos uma comparação entre ambas as línguas será justa.

Referências

1. S. Afonso, E. Bick, R. Haber, and D. Santos. Florestra sintá(c)tica: A treebank for portuguese. 2002.
2. E. Bick. *The Parsing System "PALAVRAS": Automatic Grammatical Analysis of Portuguese in a Constraint Grammar Framework*. Aarhus University Press, 2000.
3. X. Carreras and L. Màrquez. Introduction to the conll-2004 shared task: Semantic role labeling. In *Proceedings of CoNLL-2004*, 2004.
4. X. Carreras and L. Màrquez. Introduction to the conll-2005 shared task: Semantic role labeling. In *Proceedings of the Ninth Conference on Computational Natural Language Learning (CoNLL-2005)*, 2005.
5. W. Cohen. Minorthird: methods for identifying names and ontological relations in text using heuristics for inducing regularities from data, 2004.
6. Empresa Folha da Manhã S.A. Folha.com. `http://www.folha.uol.com.br`, 1921.
7. Inc. Dow Jones & Company. The wall street journal. `http://europe.wsj.com/home-page`.
8. D. Gildea and D. Jurafsky. Automatic labeling of semantic roles. *Computational Linguistics*, 28:245–288, 2002.
9. J. Hajic, M. Ciaramita, R. Johansson, D. Kawahara, M. Martí, L. Márquez, A. Meyers, J. Nivre, S. Padó, J. Štěpánek, P. Straňák, M. Surdeanu, N. Xue, and Y. Zhang. The conll-2009 shared task: syntactic and semantic dependencies in multiple languages. In *CoNLL '09 Proceedings of the Thirteenth Conference on Computational Natural Language Learning: Shared Task*, 2009.
10. P. Kingsbury and M. Palmer. From treebank to propbank. `http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.13.7566`, 2002.
11. J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of 18th International Conference on Machine Learning*, pages 282–289, 2001.
12. Linguateca. Florestra sintá(c)tica, 2009.

13. M. Marcus, B. Santorini, and M. Marcinkiewicz. Building a large annotated corpus of english: The penn treebank. *COMPUTATIONAL LINGUISTICS*, 19(2):313–330, 1993.
14. N. Miranda, R. Raminhos, P. Seabra, J. Sequeira, T. Gonçalves, and P. Quaresma. Reconhecimento de entidades nomeadas com svm. In *Actas das Jornadas de Informática da Universidade de Évora 2010*, Novembro 2010.
15. N. Miranda, R. Raminhos, P. Seabra, J. Sequeira, T. Gonçalves, and P. Quaresma. Named entity recognition using machine. 2011.
16. M. Palmer, D. Gildea, and P. Kingsbury. The preposition bank: An annotated corpus of semantic roles. *Computational Linguistics*, 31, 2005.
17. The Penn Treebank Project. The penn treebank project. `http://www.cis.upenn.edu/~treebank/`, 1999.
18. L. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. In *Proceedings of the IEEE*, pages 257–286, 1989.
19. L. Rabiner and B. Juang. An introduction to hidden markov models. *IEEE ASSP Magazine*, Janeiro 1986.
20. PÚBLICO Comunicação Social SA. Público. `http://www.publico.pt`, 1990.
21. M. Stamp. A revealing introduction to hidden markov models, 2004.
22. M. Surdeanu, R. Johansson, A. Meyers, L. Màrquez, and J. Nivre. The conll-2008 shared task on joint parsing of syntactic and semantic dependencies. In *CoNLL 2008: Proceedings of the Twelfth Conference on Natural Language Learning*, pages 159–177, 2008.
23. V. Vapnik. *Statistical Learning Theory*. Wiley-Interscience, Setembro 1998.
24. H. Wallach. Conditional random fields: An introduction. `http://citeseer.ist.psu.edu/viewdoc/summary?doi=10.1.1.124.6711`, 2004.

Analyzing the strength of ties of Retweet in health domain

Leila Weitzel[1,3], Paulo Quaresma[2] José Palazzo M. De Oliveira[3]

[1]Universidadde Federal do Pará. Campus Universitário de Marabá, Brasil.
lmartins@ufpa.br
[2]Departamento de Informática. Universidade de Évora, Portugal.
pq@uevora.pt
[3] Instituto de Informática, Universidade Federal do Rio Grande do Sul, Brasil.
palazzo@inf.ufrgs.br

Abstract. Social Network (SN) is created whenever people interact with other people. Online SN gained considerable popularity in the last years such as Facebook, Twitter and etc Twitter is SN and microblogging service that creates some interesting social network structures - *follow* relationships. Users follow someone mostly because they share common interests and they may exchange messages called *tweets*. If a user post a *tweet*, if their follower like it they repost it or retweet it. In this context, we aim to explore and study the topological structure of user's retweet network, as well, new scaling measures based on strength of retweet ties. The findings suggested that relations of "friendship" are important but not enough to find out how important users are. We uncovered other some principles that must be studied like, homophily phenomenon. Homophily explores properties of social network relationships, i.e. the preference for associating with individuals of the same background. Last but not least, it is worth emphasizing that we uncovered a weak correlation between Degree Centrality and Betweenness Centrality (49 percent) in Retweet-network and strong correlation between Degree and Betweenness centrality in Follower-network (89 percent). These find suggests that retweet network may have some fractal properties.

Keywords: Social Network Analysis, Twitter, Scaling nodes.

1 Introduction

Human beings have been part of social networks since our earliest days. We are born and live in a world of connections. Social Network is created whenever people interact, directly or indirectly, with other people. Social Networks are groups of individuals who share a commonality; they are connected by ties, or links. These links can characterize any type of relationship, e.g., friendship, authorship, etc.

Computer technologies used to create and to support social networks are relatively new. The recent proliferation of Internet Social Media applications and mobile devices has made social connections more accessible than ever before. Online Social Networks, such as Facebook, MySpace and Twitter, gained considerable popularity and grown at an unprecedented rate in the last few years [1]. Twitter is a social network-

ing and also a micro-blogging service. It creates several interesting social network structures. The most obvious network is the one created by the "*follows*" and "is *followed by*" relationships. The main goal of Twitter is to allow users to communicate and stay connected through the exchange of short messages, called *tweets*. A user posts a *tweet*, if other users like it, they repost it or *retweet* or just RT, and by a process of virality, a large number of users can be potentially reached by a particular message. The Twitter's RT capabilities can be itself useful in discovering potential relationships. Based on this context, we aim to explore and study the topological structure of user's RT network, and we propose new scaling measures based on strength of RT ties.

The outline of this paper is as follows: Section 2 we present some related works; Section 3 we explain the research methodology, data extraction technique and network modelling approach; Section 4 we provide a statistical analysis of dataset and graph analysis; and the last Section we discuss the results.

2. Related Works

One common type of social analysis is the identification of communities of users with similar interests, and within such communities the identification of the most "influential" users. Efforts have been made to measure the influence and ranking users by both their importance as hubs within their community and by the quality and topical relevance of their post. Some of these efforts are: [2–19]. Most of these researches are based on: follower, tweet and mention count, co-follower rate (ratio between follower and following), frequency of tweets/updates, who your followers follow, topical authorities. Centrality measures such as Indegree/Outdegree, Eigen Vector, Betweenness, Closeness, PageRank [20] and others have been used to evaluate node importance too. It must be stressed that, all these works are concentrated only on Twitter relationship, i.e., *follow* relationship; none of them deals with any sort of RT relationship.

3 Research Methodology

3.1 Background

Twitter let people to follow other users without approval, any user can follow you and you do not have to follow back. Thus, their ties are asymmetric and the directionality of edges are important (i.e. who is following whom) [2]. Twitter users follow someone, mostly because they are interested in the topics the user publishes in *tweets*, and they follow back because they find they share similar topic interest. These posts are brief (up to 140 characters) and can be written or received with a variety of computing devices, including cell phones. Twitter, as well as other social networks, is usually modeled as a graph **G = (V, A)** which consists of a set **V** of vertices (or nodes) representing user accounts and a set **A** of arcs (or links or ties) that connect

vertices representing relationships (*follow* relationship). Each link is an ordered pair of distinct nodes. For further details see the book "Social Network Analysis: Methods and Applications", by Wasserman and Faust, is perhaps the most widely used reference book for structural analysts [22]. The book presents a review of network analysis methods and an overview of the field.

When Twitter users are logged in, they can see a stream of tweets posted by their followers. Hence, if they like it, they can RT it, i.e., is to repeat/quote someone's tweet. The RTs posts are marked with characters RT or via @ + "screenname" in the beginning of message, we extracted either both replay tweets and mention.

- *"RT @TheNaturalNews: #Alzheimer's patients treated by playing internet games: http://t.co/dSAmzTv"*
- *"@IRememberBetter: Singing & the Brain: reflections on human capacity 4 music; pilot study of group singing w/ #Alzheimer's http://t.co/0NZXoVU #ArtAlz"*

We regarded that RT mechanism may work to increase user network in this way: a user **A** post an interesting "Tweet", you like this post and then forwarding to your network. Your followers or other user from your network discover and maybe follow the user who "Tweet", or perchance, they forward to their own network. These can potentially increasing the size and reach of user´s "Tweet" network.

3.2 Data Extraction and Modeling

We extracted the RT from 152 browsed Twitter's users; in accordance with self Twitter browse interest, in our case we selected ***health*** subject. The mining was done during March and April 2011. We crawled about 200 RT per user (this equivalent to about six month of "tweeting") totaling 4350 RT. Reference [12] demonstrated that the median number of tweets per user stay between 100 and 1000, emphasizing that maximum tweet values are closely related to the celebrities (actors, singers, pop/rock band, politicians, etc). The authors [12] proved that the majority of users who have fewer than 10 followers never tweeted or did just once and thus the median stay at 1 tweet per user. Seen this way, our sample data of RT is perfectly valid. At the end of crawling, we had a ***user-RT database*** of who replayed whom, the relationship between them and the text of retweet. At this point, we could build the RT-network. The ***RT-network*** was modeled as a direct graph G_{RT} (Figure 1) where each node $u \in V$ (totalling 1237 nodes) represents the users and each edge $a_k = (u_i, u_j) \in A$ represents RT relationship (totalling 1409 edges), i.e., an edge a_k from u_i to u_j stands that user u_i "RETWEET" user u_j . These edges a_k between nodes are weighted according the equation 1.

$$w_{a_k} = \frac{\sum \mathrm{RT}}{RT_{max}} + \alpha \qquad (1)$$

Where $\sum RT$ is the retweet count for $\boldsymbol{u}_j$, and RT_{max} is the maximum number of retweet of user $\boldsymbol{j}$. The parameter $\boldsymbol{\alpha}$ is a sort of discount rate representing Twitter relationships (follower, following, reciprocally connected and when relationships - follower or following - are absent between users). Using this notation, if an individual u_i is a "follower" of u_j , then $\boldsymbol{\alpha} \approx 0.07$ and if he/she is "following" then $\boldsymbol{\alpha} \approx 0.14$, if he/she is both follower and following then $\boldsymbol{\alpha} \approx 0.15$ and if the relationship is absent then $\boldsymbol{\alpha} \approx 0.64$. The parameter $\boldsymbol{\alpha}$ intends to discount the weight of the FOLLOW phenomenon,

since many celebrities and mass media have hundreds of thousands of followers. These values were computed according to ratio data sample.

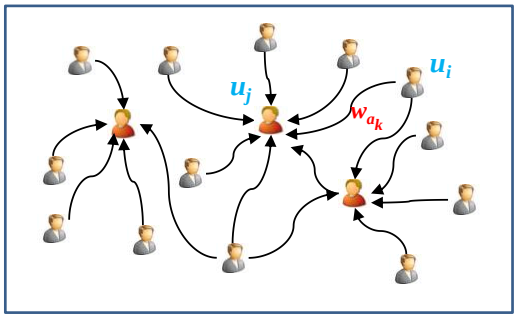


Fig. 1. Twitter RT network basic topology.

3.3 Scaling Method

F-measure is generally accepted at Information Retrieval as evaluation performance methods and by far the most widely used. It has been past more than 15 years since the F-measure was first introduced by van Rijsbergen [23]. He states, the F-measure (F) combines Recall[1] (R) and Precision[2] (P) in the following form:

$$F(R,P) = \frac{(\beta^2+1)P * R}{\beta^2 P+R} = \frac{1+\beta^2}{\frac{\beta^2}{R}+\frac{1}{P}} \quad \text{where } (0 \leq \beta \leq +\infty) \quad (1)$$

Where β is a parameter that controls a balance between P and R. When $\beta = 1$ F comes to equivalent to the harmonic mean of P and R. If $\beta > 1$, F becomes more recall-oriented and if $\beta < 1$, it becomes more precision oriented $F_0 = P$.

Each of network analysis metrics evidences a class of issue. For instance, Betweenness Centrality represents a node that occurs in many shortest paths among other nodes; this node is called "gatekeeper" between groups node. On the other hand, Closeness Centrality is the inverse of Average Distance (geodesic distance). Closeness reveals how long it takes information to spread from one node to others. Eigen Centrality measure takes into account Hub Centrality (out links) and Authority Centrality (in links). According Bonacich [21], "Eigenvector Centrality can also be seen as a weighted sum of not only direct connections but indirect connections of every length. Thus, it takes into account the entire pattern in the network. These measures are especially sensitive to situations in which a high degree position is connected to many low degree or vice-versa." Thus, at this point, we describe our approach. Let $(Rank)_i$ be the linear combination of metrics with associated weight defined by:

[1] Definition: The ratio of relevant items retrieved to all relevant items in a file [i.e., collection], or the probability given that an item is relevant [that] it will be retrieved R = (retrieved| relevant) [24].

[2] Definition: The ratio of relevant items retrieved to all items retrieved, or the probability given that an item is retrieved [that] it will be relevant P = (relevant| retrieved) [24].

$$(\text{Rank})_i = \frac{\sum_{i=1}^{n} w_i}{\sum_{i=1}^{n} \frac{w_i}{x_i}} \qquad (2)$$

Where the parameters $\sum_{i=1}^{n} w_i = (\alpha + \beta + \theta + \gamma = 1)$ are the "control balance" and used in the same way as in F-measure and $\boldsymbol{x_i}$ a set of measures: **BC** is Betweenness Centrality, **CC** is Closeness Centrality, **EC** is Eigen Vector Centrality, and **PRANK** is the PageRank [20]. We propose a set of strategic guidelines. The first proposition is that the measures have same weight (0.25), afterward each of measures is balanced according do Table 1.

Table 1. Weighted parameter: five rank approaches

Measure / Weight	α	β	θ	γ
Equal weighted	0.25	0.25	0.25	0.25
BC weighted	0.7	0.1	0.1	0.1
CC weighted	0.1	0.7	0.1	0.1
EC weighted	0.1	0.1	0.7	0.1
Prank weighted	0.1	0.1	0.1	0.7

4 Graph Analysis

An exploratory data analysis was performed to provide an overview of the available dataset. The data examination process addresses two segments: (1) a graphical examination and normality testing and (2) ranked lists analysis.

Of all 100 extracted users, only 39% did not retweet; parcels of them, 28% are mass media (newspapers, magazines, television channels and etc). This suggests that they are "traditional information provider" therefore, is expected that they not replay. We performed a "Kurtosis Normality test" and the sample passed at 95% confidence level, which allows us to state that no significant departure from normality was found. The sample of RT has a mean of 3.035 and standard deviation of 15.23. Approximately 65% had only one RT, the remaining was split between 2 and 523 retweets. The Density is low, i.e., do not have a dense "in" and "out" ties to one another. In contrast, a higher density score reflects more ties, which is generally interpreted as more coordinate network with more opportunities for sharing of information among nodes. This indicates that maybe exist potentials relationships. Conversely, Fragmentation shows that nodes are highly connected, as pointed out in Table 2 by Isolate Count Measure. The Transitivity represents the idea: "if friends of my friends are my friends", it is not quite the reality at RT network. That can be confirmed by low value of transitivity measure, see Table 2.

Table 2. RT graph-level measures.

Measures [min =0; max =1]	Values
Density	0.0009
Fragmentation	0.2567

Efficiency (the degree to which each component in a network contains the minimum links possible to keep it connected.)	0.063
Isolate Count (The number of isolate nodes in a unimodel network)	0.000
Transitivity (The percentage of link pairs {(i,j), (j,k)} in the network such that (i,k) is also a link in the network.)	0.070

We associate each position (the top 20) with a value following this approach: the first top position received 20 points, the second position nineteen, and successively decrease one unity until the last one, that received one point. Then, we compute the sum of all nodes individually for each rank approaches in Table 1 and the results of the recurring top 20 are displayed in Figure 2.

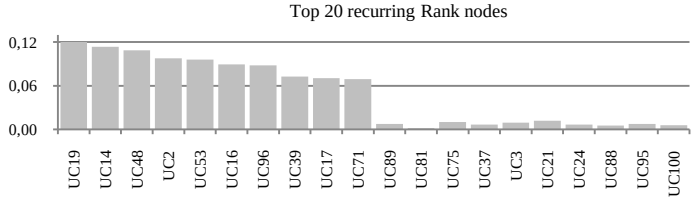


Fig. 2. Bar chart of recurring top 20 nodes.

Rather than evaluating the values calculated directly, we compute the Kendall Tau (τ) Correlation and Spearman-Rho Rank test (Rho =ρ) for the five approaches rank. The Kendall Tau ($\boldsymbol{\tau}$) Correlation and Spearman-Rho Rank test (Rho =$\boldsymbol{\rho}$) are the two most commonly used nonparametric measures of association for two random variables [24]. The correlation is significant at the 0.01 level (2-tailed) and displayed in Table 3. It is interesting to notice that all rank approaches have just same correlation and show strong positive correlation.

Table 3. Correlation matrix.

		Equal Weighted	BC Weighted	CC Weighted	EC Weighted	PRANK Weighted
Kendall's tau_b	Equal Weighted	100,00%	99,90%	99,90%	99,90%	99,90%
	BC Weighted	99,90%	100,00%	99,90%	99,90%	99,80%
	CC Weighted	100,00%	99,90%	100,00%	99,90%	99,90%
	EC Weighted	100,00%	99,90%	99,90%	100,00%	99,90%
	PRANK Weighted	99,90%	99,80%	99,90%	99,99%	99,90%
Spearman's rho	Equal Weighted	99,90%	99,90%	99,90%	99,90%	99,90%
	BC Weighted	99,90%	100,00%	99,90%	99,90%	99,90%
	CC Weighted	99,90%	99,90%	100,00%	99,90%	99,90%
	EC Weighted	99,90%	99,90%	99,90%	99,90%	99,90%
	PRANK Weighted	99,90%	99,90%	99,90%	99,90%	99,90%

The Table 4 shows the profile of top10 node according the Equal weighted rank approach. They are mainly Public Health Agencies at USA. Considering some inaccuracy about time registration, it can be seen that all of them are situated at Eastern Time (US and Canada).

Table 4. Top 10 rank nodes for Equal weighted rank approach.

ID	Followed	Followers	Tweets	Time	Joined Twitter Date
UC12	180	457	328	Eastern Time (US & Canada)	07/10/2009
UC16	28	6900	226	Quito	22/06/2010
UC14	31	116129	511	Eastern Time (US & Canada)	24/07/2008
UC17	82	1259595	414	Eastern Time (US & Canada)	28/01/2009
UC19	78	27599	797	Eastern Time (US & Canada)	21/05/2010
UC39	269	111390	1341	Quito	09/08/2007
UC48	2303	124803	2975	Eastern Time (US & Canada)	26/03/2009
UC53	92	88600	599	Quito	05/06/2009
UC71	95	4789	524	Eastern Time (US & Canada)	19/03/2009
UC96	1095	174651	2217	Eastern Time (US & Canada)	30/05/2007

5 Discussion

We proposed a new social network topological structure based on RT weighted ties to rank user influence named RT-network. We have analyzed the power of retweeting and we also have presented a new methodology to rank nodes based on control weighted parameters. The method was anchored in F-measure to control the weight balance. The experimental results offered an important insight of the relationships among Twitter users. The findings suggested that relations of "friendship" (i.e., users that have reciprocal relationship) are important but not enough to find out how important nodes are. We uncovered other some principles that must be studied like, homophily phenomenon. Homophily explores properties of social network relationships, i.e. the preference for associating with individuals of the same background. Last but not least, it is worth emphasizing that we uncovered a weak correlation between Degree Centrality and Betweenness Centrality (49 percent) in ***RT-network*** and strong correlation between Degree and Betweenness centrality in ***Follower-network*** (89 percent). References [25], show that the correlation between Degree and Betweenness Centrality of nodes is much weaker in fractal network models compared to non-fractal models. In this way, in future work we will be conduct an in-depth assessment of fractal properties in order to figure out fractal properties such as self-similarity and how to calculate their fractal dimension.

References

[1] W. Kim, O. Jeong, e S. W. Lee, "On social Web sites", *Information Systems*, vol. 35, nº. 2, p. 215-236, abr. 2010.

[2] P. Balkundi e M. Kilduff, "The ties that lead: A social network approach to leadership", *The Leadership Quarterly*, vol. 16, nº. 6, p. 941-961, dez. 2005.

[3] J. Bar-Ilan e B. C. Peritz, "A method for measuring the evolution of a topic on the Web: The case of 'informetrics'", *Journal of the American Society for Information Science and Technology*, vol. 60, nº. 9, p. 1730-1740, 2009.

[4] Bongwon Suh, Lichan Hong, P. Pirolli, e E. H. Chi, "Want to be Retweeted? Large Scale Analytics on Factors Impacting Retweet in Twitter Network", in *Social Computing (SocialCom), 2010 IEEE Second International Conference on*, 2010, p. 177-184.

[5] D. Boyd, S. Golder, e G. Lotan, "Tweet, Tweet, Retweet: Conversational Aspects of Retweeting on Twitter", in *Hawaii International Conference on System Sciences*, Los Alamitos, CA, USA, 2010, vol. 0, p. 1–10.

[6] M. Cha, H. Haddadi, e P. K. Gummadi, "Measuring User Influence in Twitter: The Million Follower Fallacy", in *International Conference on Weblogs and Social Media*, 2010.

[7] D. Gayo-Avello, "Detecting Important Nodes to Community Structure Using the Spectrum of the Graph", *Cornell University Library*, 06-abr-2010.

[8] D. Gayo-Avello, "Nepotistic relationships in twitter and their impact on rank prestige algorithms", *Arxiv preprint arXiv:1004.0816*, 2010.

[9] D. Gruhl, R. Guha, D. Liben-Nowell, e A. Tomkins, "Information diffusion through blogspace", in *Proceedings of the 13th international conference on World Wide Web*, New York, NY, USA, 2004, p. 491-501.

[10] M. Nagarajan, H. Purohit, e A. Sheth, "A Qualitative Examination of Topical Tweet and Retweet Practices", in *ICWSM 2010*, Washington, DC, 2010.

[11] F. Nagle e L. Singh, "Can Friends Be Trusted? Exploring Privacy in Online Social Networks", in *2009 International Conference on Advances in Social Network Analysis and Mining*, Athens, Greece, 2009, p. 312-315.

[12] A. Pal e S. Counts, "Identifying topical authorities in microblogs", in *Proceedings of the fourth ACM international conference on Web search and data mining*, Hong Kong, China, 2011, p. 45-54.

[13] D. M. Romero, W. Galuba, S. Asur, e B. A. Huberman, "Influence and passivity in social media", in *Proceedings of the 20th international conference companion on World wide web - WWW '11*, Hyderabad, India, 2011, p. 113.

[14] T. Sakaki e Y. Matsuo, "How to Become Famous in the Microblog World", *2010*, 2010.

[15] D. Sousa, L. Sarmento, e E. Mendes Rodrigues, "Characterization of the twitter @replies network", in *Proceedings of the 2nd international workshop on Search and mining user-generated contents - SMUC '10*, Toronto, ON, Canada, 2010, p. 63.

[16] M. J. Welch, U. Schonfeld, D. He, e J. Cho, "Topical semantics of twitter links", in *Proceedings of the fourth ACM international conference on Web search and data mining*, Hong Kong, China, 2011, p. 327-336.

[17] Y. Yamaguchi, T. Takahashi, T. Amagasa, e H. Kitagawa, "TURank: Twitter User Ranking Based on User-Tweet Graph Analysis", in *Web Information Systems Engineering – WISE 2010*, vol. 6488, Springer Berlin / Heidelberg, 2010, p. 240-253.

[18] S. Ye e S. Wu, "Measuring Message Propagation and Social Influence on Twitter.com", in *Social Informatics*, vol. 6430, Springer Berlin / Heidelberg, 2010, p. 216-231.

[19] H. Kwak, C. Lee, H. Park, e S. Moon, "What is Twitter, a social network or a news media?", in *Proceedings of the 19th international conference on World wide web*, Raleigh, North Carolina, USA, 2010, p. 591-600.

[20] L. Page, S. Brin, R. Motwani, e T. Winograd, "The PageRank Citation Ranking: Bringing Order to the Web.", Stanford InfoLab, Technical Report, 1999.

[21] P. Bonacich, "Some unique properties of eigenvector centrality", *Social Networks*, vol. 29, nº. 4, p. 555-564, out. 2007.
[22] S. Wasserman, *Social network analysis : methods and applications.*, Reprint. Cambridge: Cambridge University Press, 1999.
[23] C. J. van Rijsbergen, *Information retrieval*, 2º ed. London: Butterworths, 1979.
[24] T. Saracevic, "Evaluation of evaluation in information retrieval", in *Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval*, Seattle, Washington, United States, 1995, p. 138-146.

Marcação de Nomes Próprios usando técnicas de pesquisa local e recorrendo a fontes de conhecimento na Internet

João Laranjinho
Universidade de Évora
Évora, Portugal
joao.laranjinho@gmail.com

Irene Rodrigues
Universidade de Évora
Évora, Portugal
ipr@di.uevora.pt

Lígia Ferreira
Universidade de Évora
Évora, Portugal
lsf@di.uevora.pt

Abstract

Neste artigo apresenta-se um sistema, independente do domínio, para marcação de nomes próprio para o Português e Inglês. O sistema é avaliado de forma a estudar o impacto de diferentes fontes de conhecimento nos resultados.

O marcador usa informação morfo-sintáctica e semântica. A informação morfo-sintáctica vem de um dicionário local que completa a sua informação recorrendo a dicionários disponíveis na rede como o da Priberam e do LookWayUP. A informação semântica usada nas experiências de avaliação vem da Wikipédia e do WordNet. No sistema são usadas também algumas das técnicas pesquisa local na marcação de nomes próprios.

Na avaliação do sistema e do impacto das diferentes fontes de informação usaram-se frases de dois corpora: 100 frases com 4 nomes próprios no máximo por frase do WSJ e Brown usadas na fase de treino e 100 frases aleatórias do Brown usadas na fase de testes.

1 Introdução

O sistema que apresentamos chama-se REMUE2011. Este sistema é uma evolução do REMUE [1] que tinha como objectivo a marcação de nomes próprio para o Português. Actualmente além de marcar nomes próprios para o Português também marca nome próprios para o Inglês.

Na decisão de marcar nomes próprios usam-se dois tipos de fontes de conhecimentos:

- Informação morfo-sintáctica — do dicionário local, complementado com recurso a dicionários que estão disponíveis na Web como o dicionário Priberam [1] (para o Português) e o Look Way Up [2] (para o Inglês).
- Informação semântica — de dicionários e enciclopédias como a Wikipédia [3] e o WordNet [4] que indicam se o nome próprio existe em algum contexto.

2 Arquitectura do Sistema

A arquitectura do REMUE2011 contém 4 módulos. Na Figura 1 é apresentada a arquitectura com os seus módulos: pré-processamento, análise lexical, pesquisa local e saída.

No pré-processamento separa-se o texto em frases e as frases em átomos. As frases são constituídas por átomos e os átomos por sequências de caracteres.

Na análise lexical consulta-se em dicionários *on-line* a informação morfo-sintáctico-semântica das palavras que não se encontram no dicionário local, guardando-se essa informação no dicionário local.

[1] http://www.wikipedia.org/
[2] http://lookwayup.com/free/
[3] http://www.wikipedia.org/
[4] http://wordnetweb.princeton.edu/perl/webwn?s

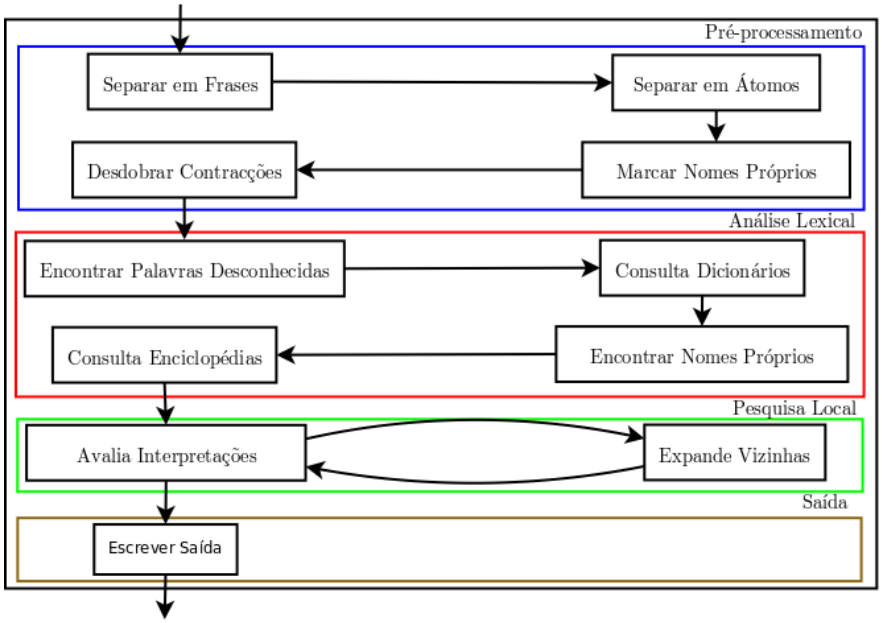


Figure 1: Arquitectura do REMUE

Na pesquisa local gera-se um conjunto de interpretações iniciais, que posteriormente serão avaliadas. Quando uma interpretação contém alterativas vizinhas com valor de heurística superior, expandem-se as vizinhas e em seguida avaliam-se. A avaliação termina quando não são encontradas mais vizinhas com valor de heurística superior.

Finalmente na saída utiliza-se a interpretação que obteve o valor mais alto de heurística.

3 Função de Avaliação

Na função de avaliação estudou-se o impacto de diferentes fontes de conhecimento na marcação de nomes próprios. O estudo das fontes incidiu sobre: o nome próprio, o átomo anterior ao nome próprio e átomo posterior ao nome próprio.

- No nome próprio estudou-se o impacto de:
  - estar na Wikipédia (WIKI), no WordNet (WORDNET) ou parte do nome na Wikipédia (WIKI_P);
  - conter maiúsculas (MAIUSCULAS) ou números (NUMEROS);
  - conter adjectivos(ADJS), advérbios (ADVS), conjunções (CONJS), determinantes (DETS), nomes comuns (NOME), preposições (PREPS), pronomes (PRONS), verbos (VERBOS) ou palavras que não estão no dicionário (DESCS);
  - o números de átomos (ATOMOS);
- No átomo anterior e posterior ao nome próprio estudou-se o impacto de:
  - estar na Wikipédia (WIKI) ou WordNet (WORDNET);
  - conter maiúsculas (MAIUSCULAS) ou números (NUMEROS);
  - ser adjectivo (ADJ), advérbio (ADV), conjunção (CONJ), determinante (DET), nome comum (NOME), preposição (PREP), pronome (PRON), verbo (VERBO) ou uma palavra que não está no dicionário (DESC);

Para optimizar os parâmetros e marcar os nomes próprios foi usada a função heurística:

$H(I) = P1 * NP\_I(I) + \sum_{i=0}^{i} (P2 * (WIKI(i) + WIKI\_P(i) + WORDNET(i) + NUMEROS(i)) * ATOMOS(i) + P3 * MAIUSCULAS(i) + P4 * (ADVS(i) + CONJS(i) + DETS(i) + PREPS(i) + PRONS(i)))$

Esta função foi encontrada realizando um conjunto de testes no qual se verificou a importância da combinação das fontes de conhecimento.

A função tem os seguintes quatro parâmetros: P1, P2, P3 e P4. Para um determinado corpora existem valores para estes parâmetros que maximizam o valor da função heurística.

Estes parâmetros serão apurados por uma aplicação (optimizador) que determina automaticamente valores para cada um dos parâmetros num deternimando intervalo.

4 Marcador Local

O marcador de nomes próprios foi criado para analisar todas as interpretações de uma frase. No entanto, em presença de frases com muitas interpretações torna-se impraticável analisar todas devido ao elevado tempo de processamento.

Para resolver este problema pensou-se em aplicar técnicas de pesquisa local no marcador, sendo um estado (interpretação) constituído por um conjunto de tuplos em que cada tuplo consiste de um átomo (palavra, sinal de pontuação, números, etc) e um valor numérico (0 ou 1) que indica se o átomo pertence ou não a um nome próprio, e um estado vizinho uma interpretação que é gerada a partir de uma outra através da mudança de um dos valores numéricos (mutação) de um tuplo ou de vários (mutações).

Para encontrar a melhor interpretação de uma frase utiliza-se o algoritmo *search_best_interpretation*:

```
interpretation function search_best_interpretation(interpretation s,number Max_Flips,table t)
  interpretation b, i, i1
  b ← s
  for j := 1 to Max_Flips do
    i ← random_interpretation(s)
    insert(i, t)
    i1 ← neighbor_interpretation(i,t)
    while i1! = null do
      if h(i1) >= h(i)
        i ← i1
        i1 ← neighbor_interpretation(i1,t)
        insert(i1, t)
      else
        i1 ← neighbor_interpretation(i,t)
        insert(i1, t)
      endif
    endwhile
    if h(i) >= h(b)
      b ← i
    endif
  endfor
  return b
end search_best_interpretation
```

O algoritmo *search_best_interpretation* recebe uma frase sem nomes próprios marcados, o número de interpretações iniciais que deve gerar aleatoriamente e uma tabela onde guarda as interpretações que gera.

Para cada interpretação inicial são expandidas as interpretações vizinhas com valores heurística superior e por sua vez as vizinhas das vizinhas.

Quando todas estas interpretações forem analisadas retorna-se a interpretação que obteve o valor mais alto de heurística.

5 Optimizador Local

Para não se apurar manualmente os valores dos parâmetros da função heurística consoante o corpora que se pretende analisar, criou-se uma aplicação que automaticamente encontra valores para os parâmetros que maximizam o desempenho da função heurística.

A aplicação foi criada com o intuito de estudar todos os conjuntos de parâmetros. No entanto, verificou-se que aumentando o número de parâmetros não se conseguia estudar todos esses conjuntos devido ao tempo de processamento.

Para contornar esse problema pensou-se em incorporar na aplicação técnicas de pesquisa local, sendo um estado constituído por um conjunto de parâmetros em que cada parâmetro é representado por um valor numérico e um estado vizinho um novo conjunto de parâmetros que contém uma diferença (mutação) ou várias (mutações) em relação ao estado anterior.

Para apurar os parâmetros utiliza-se o algoritmo *determine_parameters*:

```
parameters function determine_parameters(parameters s,number Max_Flips, table t)
  parameters b, ps, ps1
  b ← s
  for j := 1 to Max_Flips do
    ps ← random_parameters(s)
    insert(ps,t)
    ps1 ← neighbor_parameters(ps,t)
    while ps1! = null do
      insert(ps1,t)
      if ht(ps1) >= h(ps)
        ps ← ps1
        ps1 ← neighbor_parameters(ps1,t)
      else
        ps1 ← neighbor_parameters(ps1,t)
      endif
    endwhile
    if ht(ps) >= ht(b)
      b ← ps
    endif
  endfor
  return b
end determine_parameters
```

O algoritmo *determine_parameters* recebe um conjunto de parâmetros vazio, o número de conjuntos iniciais de parâmetros que pode gerar aleatoriamente e uma tabela onde guarda os conjuntos de parâmetros que gera.

Para cada conjunto de parâmetros inicial são expandidos os conjuntos de parâmetros vizinhos que dão valor de heurística superior ao texto e por sua vez os vizinhos dos vizinhos.

Quando todos os conjuntos de parâmetros forem analisados é retornado o conjunto que deu o valor mais alto de heurística ao texto.

6 Avaliação

No marcador foram estudadas duas metodologias. Uma das metodologias usa a noção de interpretação de frase e a outra apenas a noção de nome próprio.

Inicialmente na avaliação estudou-se o impacto de isolar cada uma das fontes de conhecimento com cada uma das metodologias.

Na tabela 1 podem ver-se os valores alcançados nas métricas de precisão, cobertura e medida-F quando se isolou cada uma das fontes de conhecimento e se usou a noção de interpretação.

	Nome Próprio			Átomo Anterior			Átomo Posterior		
	Prec	Cob	Med-F	Prec	Cob	Med-F	Prec	Cob	Med-F
P1*WIKI	0,3784	0,4219	0,3990	0,3454	0,1833	0,2395	0,2950	0,1858	0,2280
P1*WIKI_P	0,3987	0,4486	0,4222	X	X	X	X	X	X
P1*WORDNET	0,5244	0,3090	0,3889	0,7053	0,1365	0,2288	0,6192	0,1035	0,1774
P1*MAIUSCULAS	0,5973	0,7313	0,6575	0,3300	0,1812	0,2339	0,2975	0,1650	0,2123
P1*NUMEROS	0,8261	0,2241	0,3525	0,9908	0,0108	0,0214	0,8932	0,0142	0,0279
P1*ADJS	0,5375	0,1748	0,2638	0,5288	0,1607	0,2465	0,6433	0,1294	0,2155
P1*ADVS	0,6208	0,0248	0,0478	0,7193	0,1573	0,2581	0,8702	0,0945	0,1705
P1*CONJS	1,0000	0,0000	0,0000	0,7690	0,1165	0,2023	0,8550	0,0892	0,1615
P1*DETS	1,0000	0,0000	0,0000	0,5681	0,2053	0,3016	0,9025	0,0225	0,0439
P1*NOMES	0,5856	0,6543	0,6180	0,4179	0,2123	0,2815	0,2792	0,2537	0,2658
P1*PREPS	1,0000	0,0000	0,0000	0,5550	0,1940	0,2875	0,6708	0,2018	0,3102
P1*PRONS	0,1818	0,0868	0,1175	0,8958	0,0847	0,1547	0,8235	0,0413	0,0787
P1*VERBOS	0,6717	0,1647	0,2645	0,7274	0,0950	0,1681	0,5521	0,2778	0,3696
P1*DESCS	0,6653	0,1962	0,3030	0,6333	0,1495	0,2419	0,5119	0,3358	0,4056
P1*ATOMOS	0,5766	0,6801	0,6241	X	X	X	X	X	X

Table 1: Isolar fontes de conhecimento com noção de interpretação

Na tabela 2 podem ver-se os valores alcançados nas métricas de precisão, cobertura e medida-F quando se isolou cada uma das fontes de conhecimento e se usou a noção de nome próprio.

Nestes dois testes 1 e 2, os parâmetros foram optimizados com o marcador global no intervalo I = [-10,10] usando no treino 100 frases retiradas dos corpora Brown e WSJ com 4 nomes próprios no máximo por frase e testou-se em 100 frases aleatórias do corpora Brown.

Usando a noção de interpretação e a noção de nome próprios os resultados foram praticamente semelhantes.

As fontes que apresentaram maior impacto nos nomes próprios nas duas metodologias foram: a entrada dos nomes em enciclopédias (no caso a Wikipédia e o WordNet), a presença de maiúsculas e números nos nomes, o comprimento dos nomes e a existência de átomos nos nomes que pertencem as classes gramáticais: nome comum, adjectivo, verbo ou palavra que não se encontra no dicionário.

No átomo anterior ao nome próprio nas duas metodologias as fontes que tiveram mais impacto foram o átomo pertencer a uma das classes gramaticais: determinante, preposição, nome comum, advérbio ou adjectivo. Normalmente, palavras destas classes gramaticais antecentem os nomes próprios.

No átomo posterior ao nome próprio nas duas metodologias as fontes que tiveram mais impacto foram o átomo pertencer a uma das classes gramaticais: verbo, preposição, nome comum, advérbio ou adjectivo. Palavras destas classes gramaticais precedem os nomes próprios com bastante frequência.

Em relação às palavras que não se encontram no dicionário não podemos especular muito sobre estas, porque existe uma frequência baixa destas palavras nos corpora analisados.

	Nome Próprio			Átomo Anterior			Átomo Posterior		
	Prec	Cob	Med-F	Prec	Cob	Med-F	Prec	Cob	Med-F
P1*WIKI	0,3696	0,6029	0,4583	0,1560	0,0104	0,0195	0,1454	0,0489	0,0732
P1*WIKI_P	0,3792	0,8991	0,5334	X	X	X	X	X	X
P1*WORDNET	0,5255	0,2519	0,3406	0,6791	0,0726	0,1311	0,5631	0,0440	0,0816
P1*MAIUSCULAS	0,3093	0,8710	0,4565	0,1920	0,0673	0,0997	1,0000	0,0000	0,0000
P1*NUMEROS	0,8544	0,1893	0,3100	1,0000	0,0000	0,0000	1,0000	0,0000	0,0000
P1*ADJS	0,5447	0,2594	0,3515	0,8510	0,1128	0,1992	0,7322	0,0843	0,1512
P1*ADVS	0,6203	0,0252	0,0484	0,8552	0,1439	0,2464	0,8887	0,0672	0,1249
P1*CONJS	1,0000	0,0000	0,0000	0,8642	0,1123	0,1988	0,8653	0,0815	0,1490
P1*DETS	0,6858	0,0067	0,0132	0,6966	0,2962	0,4156	0,9000	0,0125	0,0247
P1*NOMES	0,3662	0,8751	0,5163	0,8912	0,1070	0,1911	0,5592	0,2188	0,3145
P1*PREPS	1,0000	0,0000	0,0000	0,6772	0,2500	0,3652	0,7033	0,1684	0,2718
P1*PRONS	1,0000	0,0000	0,0000	0,9150	0,0780	0,1437	0,8250	0,0173	0,0340
P1*VERBOS	0,6750	0,2299	0,3430	0,9258	0,0503	0,0955	0,6430	0,1883	0,2912
P1*DESCS	0,7086	0,2444	0,3635	0,2005	0,1752	0,1870	0,2467	0,3789	0,2988
P1*ATOMOS	0,3323	1,0000	0,4989	X	X	X	X	X	X

Table 2: Isolar fontes de conhecimento com noção de nome próprio

Na tabela 3 podem ver-se os valores das métricas obtidos com os marcadores global e local usando os optimizadores global e local com a metodologia que usa a noção de interpretação.

Na figura 2 podem ver-se os resultados dos marcadores global (GG) e local (GL) quando se usou o optimizador global.

Na figura 3 podem ver-se os resultados dos marcadores global (LG) e local (LL) quando se usou o optimizador local.

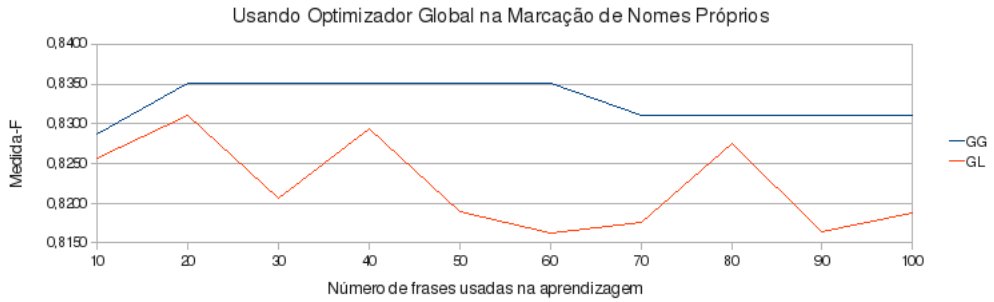


Figure 2: Usando optimizador global nos marcadores global e local

Os resultado da tabela 3 mostra que as diferenças na métrica de medida-F entre os marcadores local e global usando os optimizadores local e global não são significativas, existindo no máximo cerca de 0,03 valores de diferença na métrica de medida-F.

Nestes testes no marcador local foram usados 40 estados iniciais e 20 mutações no máximo e no optimizador local foram usados 1000 estados iniciais e 200 mutações no máximo.

Na tabela 4 e na figura 4 pode ver-se os resultados do marcador global usando o optimizador global com as duas métricas.

A métrica que usa a noção de interpretação teve resultados mais elevados (cerca de 0,20 na medida-F) que a métrica que usa a noção de nome próprio.

	MARC									
	GLOBAL				LOCAL					
FRASES	Prec	Cob	Med-F	Corr	Prec	Cob	Med-F	Corr		
10	0,7850	0,8773	0,8286	52	0,7855	0,8703	0,8257	53	GLOBAL	OPT
20	0,7889	0,8872	0,8351	54	0,7893	0,8776	0,8311	55		
30	0,7889	0,8872	0,8351	54	0,7748	0,8722	0,8206	51		
40	0,7889	0,8872	0,8351	54	0,7869	0,8768	0,8294	52		
50	0,7889	0,8872	0,8351	54	0,7764	0,8664	0,8189	50		
60	0,7889	0,8872	0,8351	54	0,7745	0,8628	0,8162	48		
70	0,7855	0,8822	0,8311	53	0,7718	0,8690	0,8175	48		
80	0,7855	0,8822	0,8311	53	0,7873	0,8719	0,8275	50		
90	0,7855	0,8822	0,8311	53	0,7748	0,8628	0,8165	46		
100	0,7855	0,8822	0,8311	53	0,7834	0,8753	0,8268	52		
10	0,7797	0,8713	0,8230	47	0,7735	0,8528	0,8112	45	LOCAL	
20	0,7905	0,8855	0,8353	54	0,7805	0,8816	0,8279	52		
30	0,7932	0,8880	0,8379	53	0,7802	0,8743	0,8246	50		
40	0,7855	0,8822	0,8311	53	0,7742	0,8683	0,8186	51		
50	0,7994	0,8955	0,8447	53	0,7930	0,8848	0,8364	49		
60	0,7925	0,8855	0,8364	55	0,7735	0,8653	0,8168	52		
70	0,7905	0,8855	0,8353	54	0,7869	0,8752	0,8287	53		
80	0,7845	0,8822	0,8305	53	0,7719	0,8683	0,8173	51		
90	0,7932	0,8880	0,8379	53	0,7707	0,8640	0,8147	49		
100	0,7855	0,8822	0,8311	53	0,7709	0,8644	0,8150	50		

Table 3: Usando noção de interpretação nos marcadores global e local com optimizadores global e local

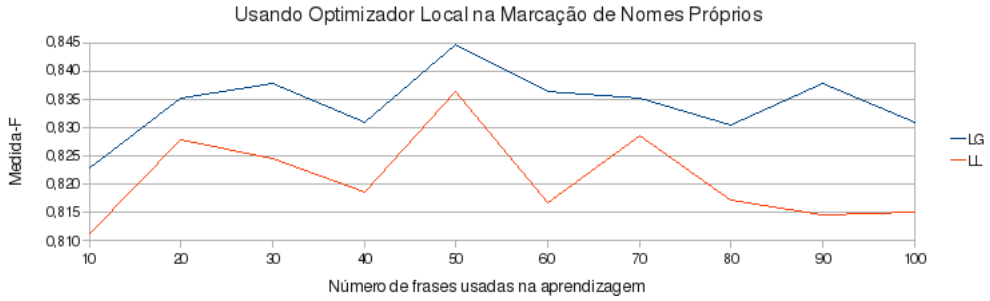


Figure 3: Usando optimizador local nos marcadores global e local

	NOÇÃO DE INTERPRETAÇÃO			NOÇÃO DE NOME PRÓPRIO		
FRASES	Prec	Cob	Med-F	Prec	Cob	Med-F
10	0,7850	0,8773	0,8286	0,6376	0,5249	0,5758
20	0,7889	0,8872	0,8351	0,6376	0,5249	0,5758
30	0,7889	0,8872	0,8351	0,5243	0,6818	0,5922
40	0,7889	0,8872	0,8351	0,5365	0,6818	0,6005
50	0,7889	0,8872	0,8351	0,5365	0,6818	0,6005
60	0,7889	0,8872	0,8351	0,5365	0,6818	0,6005
70	0,7855	0,8822	0,8311	0,5365	0,6818	0,6005
80	0,7855	0,8822	0,8311	0,5365	0,6818	0,6005
90	0,7855	0,8822	0,8311	0,5365	0,6818	0,6005
100	0,7855	0,8822	0,8311	0,5365	0,6818	0,6005

Table 4: Marcador global com optimizador global usando a noção de interpretação e nomes próprios

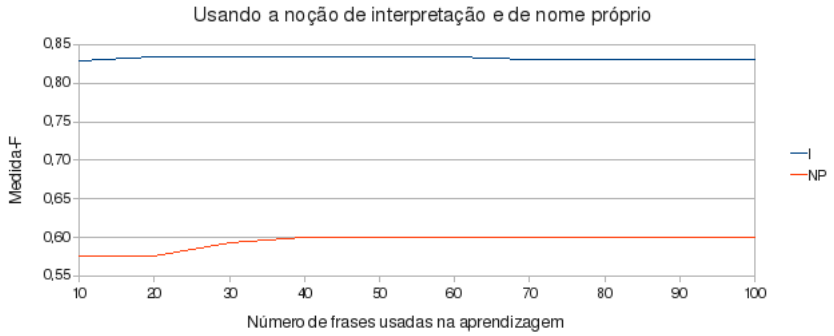


Figure 4: Usando optimizador local no marcador global com noção de interpretação e de nome próprio

7 Conclusões e Trabalho Futuro

Na marcação de nomes próprios deve explorar-se em detalhe a informação dos nomes próprios, quais as classes gramaticais dos seus átomos e se os mesmos se encontram em enciclopédias. Além disso, também é importante explorar a informação dos átomos que se encontram junto aos nomes próprios, normalmente esses átomos pertencem a um conjunto de classes gramaticais.

Nos testes efectuados, as diferenças na métrica de medida-F entre os marcadores local e global usando os optimizadores local e global não foram significativas, existindo no máximo 0,03 valores de diferença.

Ao usar o marcador local, o espaço de marcação foi reduzido em 50% e ao usar o optimizador local, o espaço de optimização foi reduzido em 99%.

Como os resultados dos marcadores local e global com os optimizadores local e global não variaram muito na medida-F (diferença máxima de 0,03), é de aprovar o uso do marcador local com o optimizador local.

O uso da noção de interpretações na marcação dos nomes próprio provou ser mais importante (cerca de 0,2 valores na medida-F) que o uso apenas da noção de nomes próprio.

Como perspectiva futura no desenvolvimento do sistema pensamos explorar a marcação de entidades mencionadas e mais tarde expandir o sistema na área análise de sentimentos.

References

[1] João Laranjinho and Irene Rodrigues. O impacto de diferentes fontes de conhecimento na marcação de nomes próprios em portugês. In *INFORUM - Simpósio de Informática*, 2010.

Using machine learning algorithms to identify named entities in legal documents: a preliminary approach

Prakash Poudyal , Luis Borrego and Paulo Quaresma

Departamento de Informática, ECT
Universidade de Évora, Portugal
{prakash,pq}@di.uevora.pt,luis_borrego@hotmail.com

Abstract. This paper deals with accuracy and performance of various machine learning algorithms in the recognition and extraction of different types of named entities such as date, organization, regulation laws and person. The experiment is based on 20 judicial decision documents from European Lex site. The obtained results were proposed for the selection of the best algorithm that selects appropriate maximum entities from the legal documents. To verify the performance of algorithm, obtained data from the tagging entities were compared with manual work as reference.

Keywords: Named entities recognition, Machine learning, Legal documents

1 Introduction

There are large scales of unstructured data stored in the web with numerous types of entities. To extract these entities from unstructured documents, information extraction algorithms are applied. However, it is quite difficult to know which algorithm is the best for particular types of entities. Therefore, it is of paramount importance to undertake experiments that could provide solutions to this problem.

The result from such study can be helpful to the lawyers, as a reference in cases when the retrieval of previous information is required. This easy way of accessing previous information may contribute towards the improvement of decision-making process.

Experiment is conducted in the default parameter of the algorithms in the minorthird [2] tool. There was no change in the parameters of the algorithms to obtain the result. Hence this experiment is given a tag of preliminary approach.

The paper is organized accordingly: in section 2 discusses related works regarding information extraction; in section 3 illustrates on concepts and tools that are

used for the experiment; in section 4 portrays on the experimental results and discussion; finally conclusion and future work is presented in the section 5.

2 Related works

Information extraction work is one of the important aspect of Machine learning: some previous works are discussion below.
The book "Knowledge Discovery from Legal Database" written by Stranieri and Zeleznikow[5] describes several approach of applying data-mining in law and also discusses trends in solving legal information extraction problem from machine learning techniques to natural language processing methodologies.

The article written by P. Quaresma and T. Goncalves [4] is the mixed approach, linguistic information and machine learning techniques to identify entities from judicial documents. Documents were available in four languages viz English, German, Italian and Portuguese. Top-level legal concepts are identified and used for document classification using support vector machine, where as named entities are identified using semantic information from output of a natural language parser.

Similarly, a book "Automatic Indexing and Abstracting of Document Texts" written by Marie-Francine Moens [3] emphasized in development of techniques for indexing and abstracting the text.

S. Baluja, V. O. Mittal and R.S.Hankar[1] presented a technique for named-entity extraction that automatically trained to recognize named-entities using statistical evidence from a training set.

3 Concepts and Tools

This section presents software used for the entity extraction, and description of dataset of judicial document.

3.1 Extraction Software

The machine-learning framework Minorthird[2]is open source software tool, which is collection of Java classes for storing text, annotating text, and learning extracting entities and categorizing text. All together 8 algorithms[1] were applied for the identification and extraction, which are listed below.

- Voted perceptron semi-Markov model (VPSMM)
- Voted perceptron conditional Markov model(VPCMM)
- Support vector machine conditional Markov model (SVMCMM)

[1] Note: Above listed algorithms are from javadoc of minorthird.

- Maximum entropy Markov model (MEMM)
- Conditional random fields (CRF)
- Semi-conditional random fields (SemiCRF)
- Voted perceptron hidden Markov model (VPHMM)
- Voted perceptron semi-Markov model 2(VPSMM2)

3.2 Dataset Description

Experiments were conducted in 20 judicial decision documents from the set of European Union law documents. These documents were obtained from EUR-Lex site[2]. The documents were available in several languages but for this experiment, English version was selected. Each document was splitted into 5 text files, which resulted in a total of 100 documents, because Minorthird suits in processing smaller text file rather than large. Entities that are extracted in this experiment listed below are:

- Name of person
- Name of organization
- Rules and Regulation Law
- Date that are available

These above entities are the most influential entities in judicial cases. The name of person, or the lawyer/criminal/judge in this case are important because they seem to appear more frequently for relevant searches, proving their influence in the related matter. The case for selection of the names of organizations, the rules and regulation laws and the dates of when the various activities occurred, a similar logic could be placed to emphasize their influence on the contextual topic. Hence these four entities have been prioritized and chosen over others. For extracting of above entities following subsets of semantic tags are given

Table 1. Entities with its semantic tags

Name of Entity	Semantic Tag
Date	<date></date>
Organization	<org></org>
Person	<person></person>
Regulation Laws	<rl></rl>

3.3 Experimental Setup

Experiments were conducted in minorthird and model was evaluated using a 10-fold stratified cross validation procedure.

[2] http://eur-lex.europa.eu/JURISIndex.do?ihmlang=en

Stratified Cross-validation: The cross-validation (CV) sometime called rotation estimation is a technique for assessing how the results of a statistical analysis will generalize to an independent data set [6]. It is a model evaluation method where the original dataset is divided randomly partitioned into k subsets (in this experiment, k=10). Then, one of the k subsets is used as the test set and the other k-1 subsets are put together to form as a training set; a model is build from the training set and then applied to the test sets. This procedure is repeated k times (one for each subset). Every data get chance to be in a test set exactly once only, and gets to be in a training set k-1 times.

Performance Measures: To know the best algorithm we analyzed precision, recall and the F_1 measure of all entities. These three terms are described briefly.

Precision is defined as the number of relevant documents retrieved divided by the total number of documents retrieved of the positive class [7].

Recall is defined as the number of relevant documents retrieved divided by the total number of elements that actually belong to the positive class [7].

For example, there is total of 9 people in the corpus and system extract only 7 of them, out of which 4 contains the names of persons and 3 of dogs. In this case precision is 4/7 and recall will be 4/9

F-measure is the harmonic mean of precision and recall and belongs to a class of functions used in information retrieval. [7] F_β can be written as

$$F_\beta = (1 + \beta^2) \cdot \frac{precision \cdot recall}{\beta^2 \cdot precision + recall}$$

4 Results and Discussion

The important two aspects are discussed here. The first is dealing with the identification of the best algorithm for each of the distinct entities and second is the comparison of the number of entities tagged by manual and machine.

Table 2 shows for each F-measure of Precision, Recall with F-measure. F-measure was considered to select the algorithm with highest value. In this case, Date has the highest value in Hidden Semi-Markov Models algorithm with f-measure value of 0.910 hence it is considered as the best algorithm but still support vector machine algorithm is competitive one with value 0.903. Similarly, Organization has highest value in Support vector machine with the value of 0.538. Similarly, Person has also highest value in Support vector machine with a 0.865. Regulation Law has highest value in Conditional Random Field with the value of 0.853.

Table 2. Precision, Recall and F measure values for each entity

	Date			Organization			Person			Regulation Laws		
	Prec	Rec	F_1	Prec	Rec	F_1	Prec	Rec	F_1	Prec	Rec	F_1
VPSMM2	.998	.096	.175	.000	.000	.000	.000	.000	.000	.000	.000	.000
VPCMM	.999	.225	.367	.446	.413	.429	.795	.339	.475	.927	.489	.640
CRF	.898	.820	.857	.659	.416	.510	.890	.840	.864	**.877**	**.831**	**.853**
SVMCMM	.898	.908	.903	**.646**	**.460**	**.538**	**.876**	**.854**	**.865**	.848	.848	.848
MEMM	.000	.000	.000	.000	.000	.000	.999	.037	.071	.000	.002	.005
SemiCRF	.000	.000	.000	.000	.000	.000	.000	.000	.000	.000	.000	.000
VPHMM	.000	.000	.000	.000	.000	.000	.967	.188	.315	.950	.495	.651
VPSMM	**.912**	**.908**	**.910**	.675	.441	.533	.803	.570	.667	.524	.055	.100

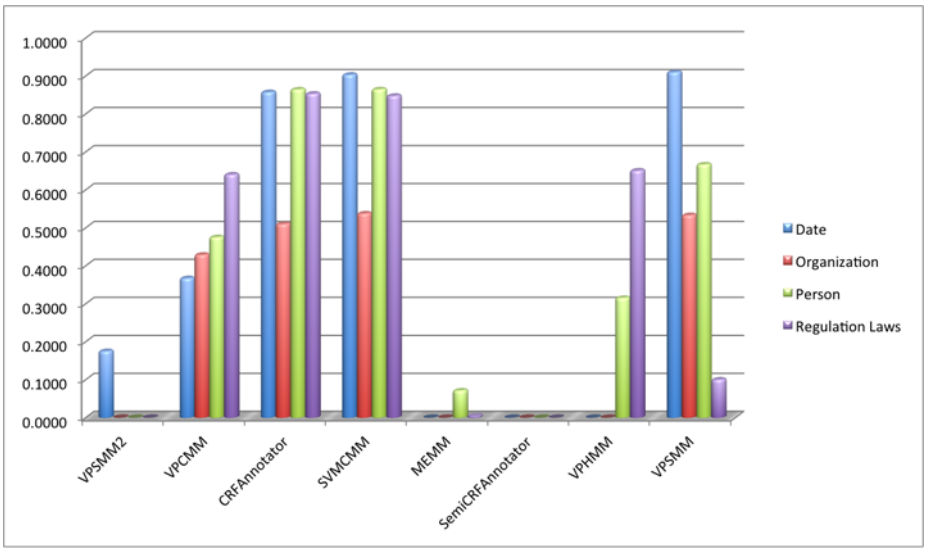


Fig. 1. bar chart of f-measure of each entities

4.1 Comparison between Manual Tagging and Machine Tagging

Manual tagging is all manual work that is conducted to tag these four entities. After conducting experiment in minorthird as explained in section 3.3; from f-measure of precision and recall, algorithm that is best to extract entities from judicial document was selected. For the verification of selecting, another setup of experiment conducted telling the respective algorithm to tag the entities in non tag judicial document, after all the result above in table 3 is more or less similar to the manual tagging number. So it can be believed by f-measure is quite promising to select the best algorithms.

Table 3. Compares manual tagging with system tagging

Entity	No. of Manual Tag	No. of Machine Tag	Algorithm	F-measure
Date	456	436	Hidden Semi Markov Model	0.910
Organization	411	395	Support Vector Machine	0.538
Person	534	531	Support Vector Machine	0.865
Regulation Laws	1388	1321	Conditional Random Field	0.853

5 Conclusion and Future work

In this paper, we have presented the results of a preliminary work aiming to automatically tag and extract information from juridical documents. The obtained results are quite promising and show that machine learning algorithms may be a good approach to deal with this problem. However, much work has to be done in order to improve the results and to be able to extract more information from the documents.

As future work, we also plan to create ontology able to represent legal knowledge and to automatically populate it with the information extracted from the legal documents.

References

1. Shumeet Baluja, Vibhu O. Mittal, and Rahul Sukthankar. Applying machine learning for high-performance named-entity extraction. *Computational Intelligence*, 16(4):586–595, 2000.
2. William W. Cohen. Methods for identifying names and ontological relations in text using heuristics for inducing regularities from data, 2004.
3. Marie-Francine Marie-Francine Moens Moens. *Automatic Indexing and Abstracting of Document Texts*, volume 6 of *The Information Retrieval*. Springer, 2000.
4. Teresa Gonçalves Paulo Quaresma. Using linguistic information and machine learning techniques to identify entities from juridical documents. 6036:44–59, 2010.
5. Zeleznikow J. Stranieri A. *Knowledge Discovery from Legal Databases. In: Law and Philosophy*, volume 69 of *Law and Philosophy Library*. Springer, Heidelberg, 2005.
6. The free encyclopedia Wikipedia. Cross-validation (statistics) - wikipedia, the free encyclopedia, November 2011.
7. The free encyclopedia Wikipedia. Precision and recall - wikipedia, the free encyclopedia, November 2011.

Framework de Pesquisas baseada numa Ontologia

Nelson Godinho, Irene Pimenta Rodrigues

Universidade de Évora
Évora, Portugal

nelson.godinho@gmail.com , ipr@di.uevora.pt

Resumo. Com o crescimento e aumento dos serviços on-line disponíveis lançados através do programa Simplex, surgiu em 2008 a primeira legislação para as Plataformas Electrónicas de Contratação Pública, cujo objectivo era desburocratizar o processo existente e, transmitir o máximo de transparência possível numa área que sempre levantou, e ainda levanta, dúvidas junto dos cidadãos nacionais.

Por estarem apenas legisladas e, por não ter sido definido um modelo de dados único que fosse usado pelas diversas plataformas existentes, cada uma criou o seu modelo sendo que a integração entre sistemas é uma tarefa difícil e dispendiosa.

Com o presente artigo pretende-se criar uma ontologia que represente um Procedimento de aquisição de bens e serviços, nomeadamente o Ajuste Directo Regime Simplificado, e uma framework de pesquisas mais eficiente baseada na ontologia criada, recorrendo a queries utilizando linguagens de interrogação para Ontologias.

Palavras-chave: ontologias, contratação pública, pesquisas semânticas, sparql

1 Introdução

Com a constante evolução tecnológica e, com o aumento de utilização por parte dos cidadãos Portugueses nos serviços *online*, por exemplo o *home banking* e as compras electrónicas, o Governo Português lançou em 2006 o SIMPLEX, um programa que consiste na Simplificação Administrativa e Legislativa que engloba um conjunto de iniciativas com o objectivo de simplificar os processos burocráticos com que os Cidadãos e Empresas se deparavam até aí.

A simplificação tem por objectivo melhorar a relação dos cidadãos com os serviços públicos, reduzir os custos de contexto das empresas no seu relacionamento com estes serviços, tornar a Administração pública mais eficiente e, assim, tornar Portugal mais competitivo. A estratégia de simplificação pode ser concretizada através de alguns objectivos genéricos:[1]

- Diminuir o número de atendimentos presenciais;
- Reduzir tempos de espera;
- Minimizar o número de interacções relacionadas com o mesmo processo;
- Prestar serviços na hora;
- Dar mais e melhor acesso à informação.

Contudo e, apesar da iniciativa do Governo ter sido criada para melhorar o dia-a-dia dos seus cidadãos, existiam diversas áreas que ainda não eram abrangidas pelo programa e que continuavam a não ser transparentes. Uma das áreas era a da Contratação Pública de serviços ou bens. Para tentar resolver este problema surgiram em 2008 os primeiros Decretos-lei para que esta área ficasse o mais transparente possível. Surgiram assim as Plataformas de Contratação Pública. Estas plataformas permitem, a realização de procedimentos electrónicos públicos, bem como a aquisição electrónica de bens e serviços.

Devido a existirem inúmeras Plataformas Electrónicas de Contratação Pública (actualmente existem em Portugal 8 Plataformas certificadas pelo CEGER[2]) foi necessário criar um Portal que reunisse toda a informação sobre os procedimentos criados em Portugal e, porque o Estado Português não quis "obrigar" os diversos organismos públicos a adoptar uma delas, surgiu assim o Portal de Contratos Públicos – BASE. No entanto, este portal apenas disponibiliza alguma das informações obrigatórias de publicação em Diário da República. Esta limitação surgiu porque cada uma das plataformas fez o seu entendimento da legislação e fez o seu próprio modelo de dados, não existindo no presente nenhum modelo único adoptado pelo estado Português.

Perante este quadro seria possível aprofundar o estudo na criação de um modelo que representasse um procedimento, vulgarmente conhecido como Concurso Público, e numa *framework* de pesquisas mais eficiente. Colocou-se o foco do artigo nestas duas áreas.

Uma das formas de conseguir este objectivo, é recorrer ao uso de ontologias, que permitem estruturar e modelar conceitos e relações de um dado domínio, fornecendo uma base de conhecimento orientada para as máquinas e respectivo processamento automático.

2 Sistema Proposto

A arquitectura do sistema proposto pode ser apresentada em duas camadas principais, uma ontológica e uma de pesquisas.

A camada ontológica pode ser vista como o *core* do sistema. É nela que está definida toda a semântica, através de uma ontologia, que representa um procedimento para aquisição de bens e serviços e as instâncias da mesma.

A camada de pesquisas fornece aos utilizadores a possibilidade de efectuarem pesquisas, neste caso, de acordo com os objectivos do trabalho efectuado, na camada ontológica. A figura 1 representa a arquitectura do sistema.

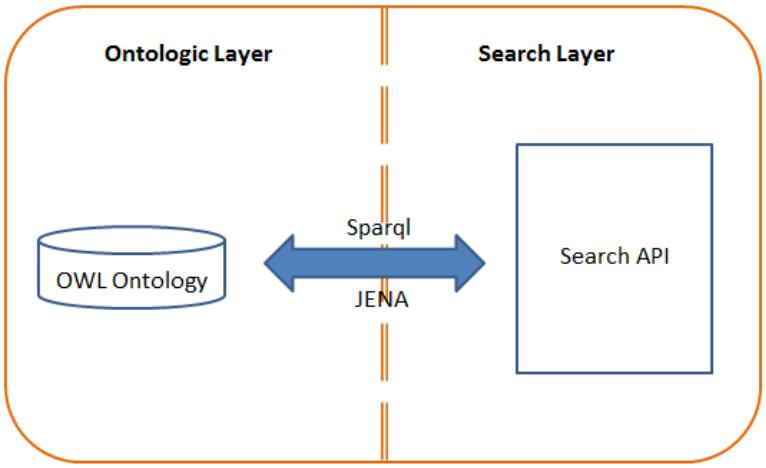


Fig. 1. Arquitectura do Sistema

Este sistema foi desenhado para responder à legislação portuguesa DL nº 18/2008 de 29 de Janeiro [3], do DL n.o 143-A/2008 [4] de 25 de Julho e da Portaria 701-G/2008 [5] de 29 de Julho, no que respeita à realização de procedimentos electrónicos públicos para aquisição de bens e serviços.

2.1 Ontologia

Como referido anteriormente, a base da camada ontológica é uma ontologia que representa um procedimento para aquisição de serviços e bens.

Para criar esta ontologia fez-se uma análise à legislação Portuguesa sobre procedimentos electrónicos de aquisição de bens e serviços e a uma plataforma electrónica de contratação pública, neste caso em particular o Bizgov.

A metodologia utilizada para criar a ontologia foi a *Simple Knowledge-Engineering Methodology* [6] e utilizou-se o OWL [7] como linguagem de representação, por ser mais eficiente e por ser a recomendação do W3C.

A ontologia criada encontra-se representada na figura 2.

Fig. 2. Hierarquia de classes

2.2 Pesquisas

Esta camada tem o propósito de fornecer aos utilizadores a possibilidade de efectuarem pesquisas na ontologia que obedeçam aos objectivos definidos para este artigo.

Para construir esta camada recorreu-se à linguagem SPARQL [8] e foi definida a seguinte query, conforme os objectivos definidos.

Objectivo: Devolver a lista de domínios (procedimentos, entidades adjudicantes e fornecedores) onde exista a palavra pesquisada, definida pela seguinte sintaxe:

```
PREFIX mydomain:
<http://localhost/Public_Contracting.owl>

SELECT DISTINCT ?proc
WHERE
{
    ?proc a mydomain:Procedure;
          mydomain:hasBuyer ?buyer;
          mydomain:hasSupplier ?supplier.
    ?supplier mydomain:hasDistrict ?supplierDistrict.
```

```
    ?buyer mydomain:hasDistrict ?buyerDistrict.
    FILTER (?supplierDistrict = 'String Pesquisada' ||
?buyerDistrict = 'String Pesquisada')
}
```

Com esta query são retornados todos os procedimentos que contenham a palavra pesquisada. Depois basta percorrer, recorrendo à framework Jena, esta lista e listar todas as entidades adjudicantes e fornecedores recorrendo às propriedades *has_buyer* e *has_supplier*.

Na figura 3 encontram-se representadas as propriedades necessárias para obter os resultados.

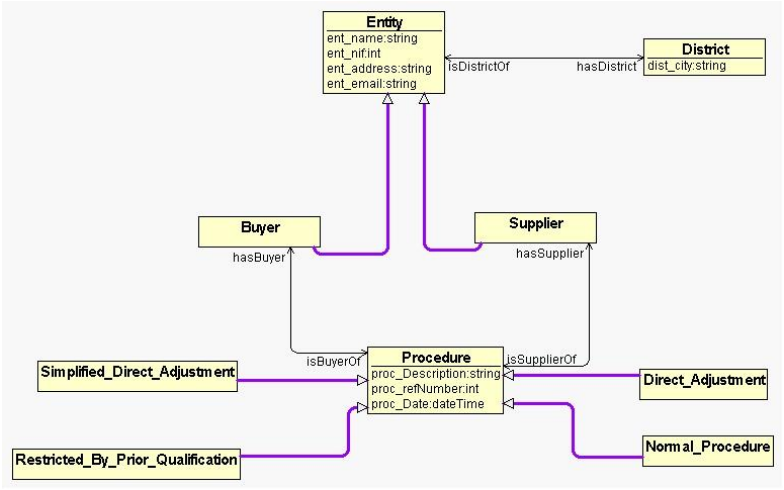


Fig. 3. Propriedades utilizadas na query de pesquisas

2.3 Testes Efectuados

Depois de ter sido definida a ontologia e respectivas instâncias e implementada a camada de pesquisas foram efectuados testes para comprovar a validade dos resultados obtidos.

Pesquisa por "Évora"

Utilizando a query definida sobre as instâncias representadas na tabela 1, e pesquisando pela string "Évora" foram obtidos os resultados enunciados na tabela 2:

Procedimento	**Entidade Adjudicante**	**Distrito da Ent. Adju.**	**Fornecedor**

Procedure 1	Ent. Adju. Teste 1	Évora	Fornecedor Teste 1
Procedure 1	Ent. Adju. Teste 1	Évora	Fornecedor Teste 2
Procedure 2	Ent. Adju. Teste 2	Évora	Fornecedor Teste 1
Procedure 3	Ent. Adju. Teste 3	Porto	Fornecedor Teste 3
Procedure 3	Ent. Adju. Teste 3	Porto	Fornecedor Teste 4
Procedure 4	Ent. Adju. Teste 4	Évora	Fornecedor Teste 1
Procedure 4	Ent. Adju. Teste 4	Évora	Fornecedor Teste 4
Procedure 4	Ent. Adju. Teste 4	Évora	Fornecedor Teste 5
Procedure 5	Ent. Adju. Teste 5	Faro	Fornecedor Teste 6
Procedure 5	Ent. Adju. Teste 5	Faro	Fornecedor Teste 7

Tabela 1. Instâncias criadas para testes

Procedimento	**Entidade Adjudicante**	**Distrito da Ent. Adju.**	**Fornecedor**
Procedure 1	Ent. Adju. Teste 1	Évora	Fornecedor Teste 1
Procedure 1	Ent. Adju. Teste 1	Évora	Fornecedor Teste 2
Procedure 4	Ent. Adju. Teste 4	Évora	Fornecedor Teste 1
Procedure 4	Ent. Adju. Teste 4	Évora	Fornecedor Teste 4
Procedure 4	Ent. Adju. Teste 4	Évora	Fornecedor Teste 5

Tabela 2. Resultados da Pesquisa por Évora

3 Conclusão

Este sistema faz uso de uma ontologia que representa um procedimento de aquisição de bens e serviços e de uma *framework* de pesquisas sobre a ontologia definida.

Tratando-se de um trabalho de investigação, por não existir nada implementado nesta área específica, foi necessário procurar várias hipóteses para se conseguir chegar à solução final. Nem sempre a solução estudada foi a melhor, sendo por isso necessário recuar e procurar novos caminhos, em parte devido a não existir uma única forma de definir ontologias.

A ontologia criada envolve um conjunto de classes, propriedades e restrições e está representada na linguagem OWL. Esta ontologia foi criada de raiz, pois como referido anteriormente, não existem trabalhos desenvolvidos neste âmbito.

A *framework* de pesquisas foi pensada para responder a questões que não são possíveis na plataforma de contratação pública actual. Estas questões são uma mais valia para os seus futuros utilizadores, visto que até aqui apenas era possível pesquisar

por procedimentos ou por fornecedores. Com a solução proposta, os utilizadores conseguirão pesquisar por um domínio e obter resultados indexados a vários recursos como por exemplo, procedimentos, entidades adjudicantes e fornecedores. Esta *framework* foi implementada recorrendo ao Sparql e/ou às funcionalidades fornecidas pelo Jena.

Referências

1. Presidência do Concelho de ministros. Orientações para a simplificação, 2009. Disponível em: http://www.simplex.pt/downloads/orientacoesideiasimplex.pdf. Último acesso: 14/08/2011.
2. Base Portal de Contratos Públicos Online. Plataformas de contratação pública certificadas, 2011. Disponível em: http://www.base.gov.pt/plataformaselectronicas/Paginas/plataformascertificadas.aspx. Último acesso: 14/08/2011.
3. Decreto-lei nº 18/2008, 2008. Disponível em: http://dre.pt/pdf1s/2008/01/02000/0075300852.pdf. Último acesso: 14/08/2011.
4. Decreto-lei nº 143-a/2008, 2008. Disponível em: http://dre.pt/pdf1sdip/2008/07/14301/0000200006.pdf. Último acesso: 14/08/2011.
5. Portaria 701-g/2008, 2008. Disponível em: http://www.base.gov.pt/codigo/Documents/Portaria701G2008.pdf. Último acesso: 14/08/2011.
6. Natalya F. Noy and Deborah L. Mcguinness. Ontology Development 101: A Guide to Creating Your First Ontology. Online, 2001. Disponível em: http://www.ksl.stanford.edu/people/dlm/papers/ontology101/ontology101-noy-mcguinness.html. Último acesso: 20/10/2011.
7. Deborah L. McGuinness and Frank van Harmelen. . web ontology language - overview. w3c recommendation, 2004. Disponível em: http://www.w3.org/TR/owl-features. Último acesso: 12/06/2011.
8. W3C. Sparql query language for rdf, 2008. Disponível em: http://www.w3.org/TR/rdf-sparql-query/. Último acesso: 16/08/2011.

Das Bases de Dados Prosopográficas à Análise de Redes: Ensaios de Aplicação a Dados Históricos

Albertina Ferreira[1] Carlos Caldeira[2] Fernanda Olival[3]

[1] Instituto Politécnico de Santarém; email: albertina.ferreira@esa.ipsantarem.pt
[2] Universidade de Évora; email: ccaldeira@di.uevora.pt
[3] Universidade de Évora; email: mfo@uevora.pt

Resumo As redes sociais são actualmente uma realidade incontornável e abrangente, fundamentais para melhor compreender a sociedade. Este trabalho pretende, em termos globais, iniciar uma análise das relações existentes entre os diversos protagonistas dos processos de Familiaturas do Santo Ofício. Em termos concretos, aborda-se a investigação e o conhecimento reunidos no âmbito das redes, ontologias e *data mining*. Por um lado, focalizamo-nos no interesse suscitado pela teoria dos grafos, com destaque para as medidas de centralidade, salientando a importância da visualização das redes e das várias dinâmicas sociais. Por outro lado, referimos a importância do desenvolvimento de ontologias e da aplicação de *data mining* às redes sociais. No ensaio preliminar aplica-se uma das medidas de centralidade, o grau, ao nosso caso de estudo.

1 Introdução

Durante os últimos anos, o estudo de teorias de redes nas ciências físicas e sociais tem sido uma área pela qual os investigadores demonstram grande interesse. Para os cientistas sociais, a teoria das redes tem possibilitado explicações para os fenómenos sociais, numa ampla variedade de disciplinas que vão desde a psicologia à economia [3].
Newman *et al.* [9] comentam que as redes estão em toda parte e que problemas dinâmicos estão na vanguarda da pesquisa em rede, onde há muitas perguntas ainda sem resposta. Mais recentemente Lazer *et al.* [8] referem que vivemos a vida em rede.
Embora os autores anteriormente focados considerem essencialmente redes a funcionar na actualidade, grande parte dos estudos que realizam poderão ser estendidos a outras épocas, bem como a outras sociedades. Neste trabalho propomo-nos estudar de que modo se relacionavam os diversos intervenientes nos processos de Familiaturas do Santo Ofício. Este estudo será enquadrado no âmbito mais lato das redes sociais e desenvolvido no âmbito do projecto aprovado e financiado pela FCT: PTDC/HIS-HIS/118227/2010 – Grupos intermédios em Portugal e no Império Português: as familiaturas do Santo Ofício (c. 1570-1773) – Instituição sede: CIDEHUS[4].

[4] Centro Interdisciplinar de História, Culturas e Sociedades da Universidade de Évora

2 Estado da Arte

Uma das ideias que mais prevalece nas ciências sociais é a noção de que as pessoas constituem redes que representam as relações sociais e suas interacções. Ou seja, os atributos de um conjunto de indivíduos são estáticos e insuficientes para explicar o modo como actuam e obtêm trunfos ou vantagens sociais. Deste modo, a teoria das redes sociais oferece uma resposta para uma questão que é colocada desde a época de Platão: como funciona a sociedade? A teoria de redes também fornece explicações para muitos fenómenos sociais, que vão desde a criatividade individual até à rentabilidade das empresas [3].
Para Snijders *et al.* [12], a evolução nas redes sociais é um domínio de investigação com alguma complexidade. Como é que uma rede social evolui? Podemos encontrar leis e derivar modelos que explicam a sua evolução? Como é que as comunidades surgem numa rede social? Estas questões são só um exemplo de todas aquelas que se podem colocar neste âmbito de investigação e às quais podemos acrescentar: como é que redes, onde a variação temporal é um ponto crucial, admitem cortes cronológicos e aceitam dados fragmentários?
São muitas as descrições que se encontram sobre redes sociais. Andery *et al.* [1] referem que a maneira mais usual de apresentar redes sociais é utilizando grafos. Estes são diagramas constituídos por vértices e arestas que representam os indivíduos da rede e as relações que estes mantêm entre si.
A estrutura das redes pode ser analisada utilizando medidas de centralidade: grau (número de relações que um nó estabelece com os nós que lhe são adjacentes), proximidade (distância de um nó relativamente aos restantes nós na rede), intermediação (intermediação que um nó faz com os nós que com ele não se conectam directamente) [7].
A aplicação de ontologias pode ser um caminho para melhor organizar-se a informação. Em 1993, Gruber [5] definiu ontologias como *"explicit specification of a conceptualization"*. Refere ainda que para sistemas baseados em conhecimento, o que existe é exactamente o que pode ser representado. Desde modo, descreve a ontologia de um programa a partir da definição de um conjunto de termos representativos. Embora não exista uma metodologia modelo para aplicar no desenvolvimento de uma ontologia, este autor salienta que esta deve ser clara, objectiva, coerente e extensível.
No que respeita a metodologias aplicáveis à construção de uma ontologia, Rautenberg *et al.* [11] referem cinco etapas: especificação (identificar o objectivo e o âmbito da ontologia); conceptualização (descrever um modelo conceptual); formalização (transformar o modelo conceptual num modelo formal); implementação (desenvolver a ontologia formalizada numa linguagem de representação apropriada); manutenção (actualizar e corrigir a ontologia quando necessário).
A organização da informação conseguida através de uma ontologia poderá ser valorizada, gerando conhecimento que posteriormente pode ser compartilhado, quando se utilizam técnicas de *data mining*. Através destas é possível identificar padrões e tendências em repositórios com elevado número de registos. Fayyad *et al.* [4] definem assim *data mining* como: *"... a step in the KDD process that consists of applying data analysis and discovery algorithms that, under accep-*

table computational efficiency limitations, produce a particular enumeration of patterns (or models) over the data". Estes autores referem ainda o interesse, para a procura de padrões, da aplicação de regras de classificação, regressões e análise de *clusters*. Para além destas, Han *et al.* [6] consideram também a análise de *outliers* como uma das funcionalidades de *data mining*.
Para Han *et al.* [6] a análise de *clusters* estuda e agrupa os dados que possuem características similares, permitindo a organização de observações numa hierarquia de classes que agrupam eventos semelhantes. A análise de *outliers*, referida por estes autores, investiga os dados que não apresentam um comportamento idêntico ao da maioria, revelando-se muitas vezes os eventos raros mais interessantes do que os eventos que regularmente ocorrem.

3 Procedimentos Metodológicos

O estudo em questão possui uma perspectiva longitudinal, tendo em vista que os dados a utilizar se encontram distribuídos por três séculos, recaindo o nosso estudo sobre um número bastante elevado de registos, os quais se encontram disponíveis na base de dados prosopográfica *SPARES*[5], que actualmente tem 83.163 registos. Esta base de dados foi desenvolvida por Carlos Pampulim Caldeira no âmbito do projecto FCOMP-01-0124-FEDER-007360 – Inquirir da Honra: Comissários do Santo Oficio e das Ordens Militares em Portugal (1570 – 1773).
A ferramenta escolhida para construir e analisar a rede foi o *Pajek*. Na base desta escolha esteve, por um lado, a possibilidade desta aplicação poder explorar e manipular redes de grande dimensão [10] e, por outro, correr sobre o sistema Windows. Encontra-se disponível gratuitamente, para uso não comercial, e pode ser acedido a partir de: `http://vlado.fmf.uni-lj.si/pub/networks/pajek/` [2]. É utilizando este software que iremos construir a rede e analisá-la no que diz respeito a uma das medidas de centralidade referida no capítulo 2: grau.
Por possuir uma arquitectura modular e estar em constante desenvolvimento, a ferramenta *Protégé*, disponível em `http://protege.stanford.edu`, foi o editor que escolhemos para futuramente utilizarmos na construções e desenvolvimento de ontologias. Esta ferramenta permite a implementação de diversas metodologias que possibilitam não só a definição de classes e de hierarquias como também a implementação de restrições ao nível das propriedades.
Consideramos que as tarefas de *data mining* serão fundamentais para o reconhecimento de padrões, concretamente no que respeita à análise de *clusters* e de *outliers*. Pensamos complementar a abordagem que será feita através do *Pajek* com a utilização da ferramenta *Weka*. Esta encontra-se disponível gratuitamente em `http://www.cs.waikato.ac.nz/ml/weka/`.

4 Ensaio Preliminar

Iniciou-se este ensaio com uma pesquisa à base de dados *SPARES*. Nesta trataram-se os dados gerando meta-dados, de modo que estes pudessem ser analisados no

[5] Sistema Prosopográfico de Análise de Relações e Eventos Sociais

Pajek. Utilizando esta ferramenta criou-se a rede que pode ser observada na Figura 1. A observação global desta rede não é impossível, mas, como se pode

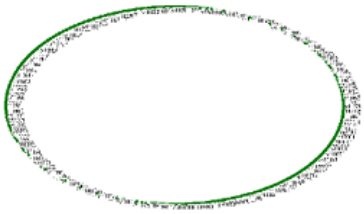


Figura 1. Rede visualizada com a visualização Layout Circular/Random.

constatar, trata-se de uma visualização a uma escala pouco esclarecedora. Continuámos este ensaio contemplando duas hipóteses. Na primeira considerámos o tipo de relação que existia entre os diversos protagonistas. Na segunda, fizemos uma restrição no que diz respeito à data inicial (este é um dado sempre conhecido, nem que seja por aproximação). Concluímos ambas as hipóteses fazendo um estudo sobre uma das medidas de centralidade anteriormente referidas: grau.

4.1 Exploração de dados por tipo de relação

Na base de dados *SPARES* encontram-se referidos diversos tipos de relações. Apresentamos neste ensaio os procedimentos realizados para uma relação de "Patrocínio" (quando um individuo intervém de forma explícita a favor de outro). Apresenta-se na Figura 2 o resultado da visualização desta relação no *Pajek*. Pode observar-se que, para este tipo de relação, cinco (72%) dos indivíduos têm

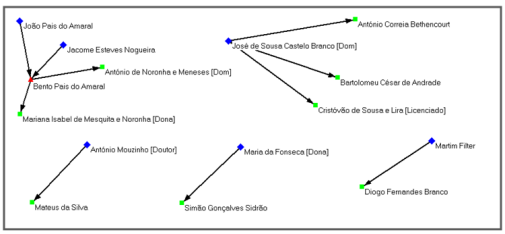


Figura 2. Rede criada no *Pajek* para uma relação de Patrocínio.

apenas um patrocínio, um (14%) indivíduo patrocina dois indivíduos e um (14%) indivíduo patrocina três indivíduos. De um total de 15 indivíduos, apenas um (Bento Pais do Amaral) patrocina e é simultaneamente patrocinado (7%).

Considerando que o grau é o número de relações de cada um dos comissários relativamente às testemunhas, detectaram-se três situações possíveis: 5 comissários possuem grau 1; 1 comissário possui grau 2; 1 comissário possui grau 3. Podemos assim constatar que a testemunha (que também exerce o papel de comissário) Bento Pais do Amaral é *prominent*, possui grande prestígio, e que o actor Dom José de Sousa Castelo Branco é *influential* e possui uma forte ascendência sobre todos os outros [7].

4.2 Exploração dos dados limitando a data inicial

Para a segunda hipótese, considerámos os processos relativos ao século XVI, no que diz respeito à relação entre comissários e testemunhas. Pode observar-se na Figura 3 o resultado visualizado no *Pajek*. Da observação destes dados podemos

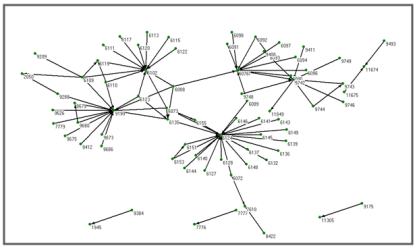


Figura 3. Rede criada no *Pajek* para uma data inicial pertencente ao século XVI.

retirar algumas ilações: existem 5 comissários que ao fazerem interrogatórios recorrem a um número diversificado de testemunhas; outros ouvem muitas vezes apenas uma mesma testemunha; na maior parte das situações, as testemunhas encontram-se apenas relacionadas com um comissário, sinal de que este ouve muitas pessoas diferentes; algumas testemunhas são simultaneamente comissários. Apresenta-se na Tabela 1 os resultados obtidos relativamente à medida de centralidade grau. Verifica-se, assim, que 67% das testemunhas encontra-se relacionada

Tabela 1. Número de testemunhas por graus

Grau	Número de testemunhas
1	51
2	8
3	2
4	1
5	2

com apenas um comissário, 11% com 2 comissários, 3% com 3 comissários, 1%

com 4 comissários e 3% com 5 comissários. Estas testemunhas totalizam 84% da rede, sendo nalgumas situações também comissários. Os restantes 16% da rede correspondem a comissários que não se encontram relacionados com nenhuma testemunha.

5 Conclusões e Trabalho Futuro

Este trabalho pretendeu iniciar-nos na análise de redes que permita estudar as relações que existiam entre os diversos protagonistas.
Como trabalho futuro, pretendemos, por um lado, produzir redes dinâmicas em função de variáveis cronológicas, uma vez que a variação temporal, bem como o carácter fragmentário dos dados são uma constante nos dados que exploramos. Por outro lado, temos como objectivo explorar o parentesco horizontal como rede, pois este conhecimento é muitas vezes mais importante do que conhecer o parentesco vertical que actualmente se encontra implementado. É também nossa intenção empregar *data mining* e ontologias para a determinação de padrões que possam ser úteis e interpretáveis. Por último, é nosso desejo encontrar alternativas de interoperabilidade entre as várias ferramentas anteriormente referidas.

Referências

1. Andery, G. F., Lopes, A. A., Minghim, R.: Exploração visual multidimensional de redes sociais. 2nd International Workshop on Web and Text Intelligence. São Carlos. (2009) 1–9
2. Batagelj, V., Mrvar, A.: Pajek: Program for Analysis and Visualization of Large Networks. Reference Manual List of commands with short explanation version 2.00. University of Ljubljana. Slovenia. (2010)
3. Borgatti, S. P., Mehra, A., Brass, D. J., Labianca, G.: Network Analysis in the Social Sciences. Science. **323** (2009) 892–895
4. Fayyad, U., Piatetsky-Shapiro, G.; Smyth, P.: From data mining to knowledge discovery: an overview. AI Magazine. **17(3):** (1996) 37–54.
5. Gruber, T.: A Translation Approach to Portable Ontology Specifications. Knowledge Acquisition. **5(2):** (1993) 199–220.
6. Han, J.; Kamber, M.; Pei, J.: Data Mining-Concepts and Techniques. Third Edition. The Morgan Kaufmann Series in Data Management Systems. Morgan Kaufmann Publishers. (2011)
7. Hanneman, R. A., Riddle, M.: Introduction to social network methods. University of California, Riverside. (2005) `http://faculty.ucr.edu/~hanneman/`
8. Lazer D. et al.: Life in the Network: the Coming Age of Computational Social Science. Science. **323** (2009) 721–723
9. Newman, M. E. J., Barabási, A., Watts, D. J.: The Structure and Dynamics of Networks. (2006) `http://press.princeton.edu/chapters/s8114.html`
10. Nooy, W, Mrvar, A, Batagelj, V.: Exploratory Network Analysis with Pajek. Cambridge University Press. New York. (2005)
11. Rautenberg, S.: Uma Metodologia para o Desenvolvimento de Ontologias. Revista Ciências Exatas e Naturais. **10(2):** (2008) 237–262
12. Snijders, T.A.B., Steglich, C.E.G., van de Bunt, G.G.: Introduction to Actor-Based Models for Network Dynamics. Social Networks. **32** (2010) 44–60

Automatic Ontology Population extracted from SAM Healthcare Texts in Portuguese

David Mendes; Irene Pimenta Rodrigues

Departamento de Informática da Universidade de Évora

{dmendes;ipr}@uevora.pt

Abstract. We describe a proposal of the steps needed to automatically extract the information about healthcare providing activities from an actual EHR[1] at use in a Portuguese Region (Portalegre) to populate an Ontology. We present the steps to manually and further automatically populate using a suggested Software Architecture and the appropriate Natural Language Processing techniques for Portuguese Clinical jargon.

1 Introduction

We will present in this paper our proposal on how to populate a clinical practice ontology, CPR[2], automatically from texts that can be obtained from the SAM [3] software system at use in ULSNA [4].

1.1 Motivation

The Semantic Web tools and techniques have come of age to be able to use an Ontology about a specific scientific and/or professional domain as knowledge representation scaffolding enough to be able to reason automatically and semantically inter-operate in that domain. The enormous amount of data residing in EHR renders the possibility of manually curating an ontology from that wealthy resource virtually impossible. We think that we dominate enough scientific knowledge about Natural Language Processing in general, and Portuguese in particular, to try applying it to the Healthcare providing domain in order to seriously contribute to the enhancement of the quality and cost effectiveness of that important activity.

1.2 Previous work

After researching the State-of-the-Art in knowledge acquisition from text in the Biomedical domain we presented some papers to peer-reviewed internacional

[1] Electronic Health Record
[2] Computer-based Patient Record Ontology
[3] Sistema de Apoio ao Médico - Doctors Support System
[4] Unidade Local de Saúde do Norte Alentejano - North Alentejo Local Health Unit

conferences about ontology enrichment in the heathcare domain. Several agreements with local heath authorities and healthcare providers have been signed in order to be able to develop work with reasonable corpora of data to demonstrate the applicability of the work of this research in real world situations.

1.3 Work in progress

We have colected some dozens of texts in PDF format that we use in this report both manually and automatically, as seen ahead, to enrich the CPR ontology. We are in the process of demonstrating the possibility of information extraction from free-text clinical episode reports to populate the ontology in an automated manner.

2 Experiences in the field

The ULSNA, E.P.E. [5] has as its principal object the provision of primary and secondary health care, rehabilitation, palliative and integrated continued care to the population. In particular to the beneficiaries of the national health service and the beneficiaries of the health subsystems, or with external entities with which it contractualized the provision of health care and to all citizens in General. Also articulate with public health activities and the means necessary to exercise the powers of the health authority in the geographic area affected by it. ULSNA, also has as it object to develop research, teaching and training activities. ULSNA is a healthcare providing regional system that includes 2 hospitals (José Maria Grande in Portalegre and Santa Luzia in Elvas) and the primary care centers in all the district counties: Alter do Chão, Arronches, Avis, Campo Maior, Castelo de Vide, Crato, Elvas, Fronteira, Gavião, Marvão, Monforte, Nisa, Ponte de Sôr, Portalegre e Sousel. Universidade de Évora signed an agreement with ULSNA that enabled the usage of de-identified (according to safe-harbor principles) clinical data from the SAM system in use both in the Primary Healthcare units and in the Hospitals. Using the clinical data that was available for us we populate the ontology for automatic reasoning capabilities over the suggested OWL2 ontology.

2.1 Acquisition points from SAM SOAP to CPR

In any EHR the number of direct sources for ontology instance retrieval is enormous. The number of registered clinical episodes from which we can populate our ontology induces any ontology engineer in a very hard problem to solve just trying to figure out the granularity of the ontology to be able to represent a realistic view from where valuable information can be extracted. When trying to apply the principles of well defined formal ontologies depicted in [5] and trying to avoid the errors mentioned in [1] we decided to get our hands wet with a

[5] www.ulsna.min-saude.pt

simple approach to the representation of disease and diagnostic as illustrated in [4].

In the SAM system there lies a clear support for text divided by 4 pre-defined subsections curiously acronymed SOAP after Subjective, Objective, Analysis and Plan. For any particular encounter (actually for any Clinical Episode) the text for any of these may be collected in the form of text suitable for processing into the Ontology and the appropriate suggested points in the CPR timeframe are the following:

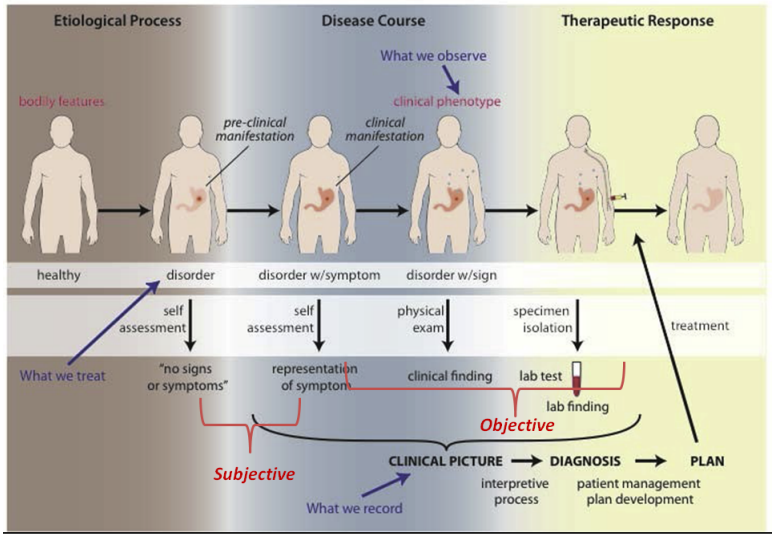


Fig. 1. SOAP Points Insertion

Where Subjective notes are those kind of signs normally expressed by the patient as well as soft validatable symptoms. Objective findings are all kinds of observations and results from quantifiable exams. Processing and populating the Ontology with the Analysis record labeled as Diagnosis the "Clinical Picture" is completed and is only lacking the Plan being instanced to render the therapeutic response associated with this encounter.

3 Automated acquisition from Clinical Episodes Text

3.1 The generic situation

As reviewed by the authors in Mendes and Rodrigues [2] the state-of-the-Art for acquisition from Clinical Text has enjoyed strong developments in recent years. In the mentioned paper we presented a proposal for automated acquisition from

HL7 messaging but here we are delving into the more generic possibility of extracting from free text present in most interfaces used by clinicians. Going from clinical episodes free text that is usually presented in a human friendly format to one adequate for computer processing involves a fair amount of text processing to handle situations like:

- Reports aggregate information from different clinical episodes that are not uniquely identified or not even individually dated
- The clinician is only identified by his/her name if any identification is made at all
- The information conveyed in free text is intended only to be understandable by fellow practitioners or even by the clinician himself making use of pragmatic jargon normally plagued with acronyms and nicknames abundant in their specific community
- Text is profoundly intermixed with decorative elements for better legibility, normally in PDF or HTML files
- The clinicians natural language is other than English without concepts defined in the foundational thesaurus like SNOMED CT or FMA for instance that don't exist in that particular language
- The time spanning of the processes depicted in natural language are difficult to represent formally

3.2 The adequate annotation workflow

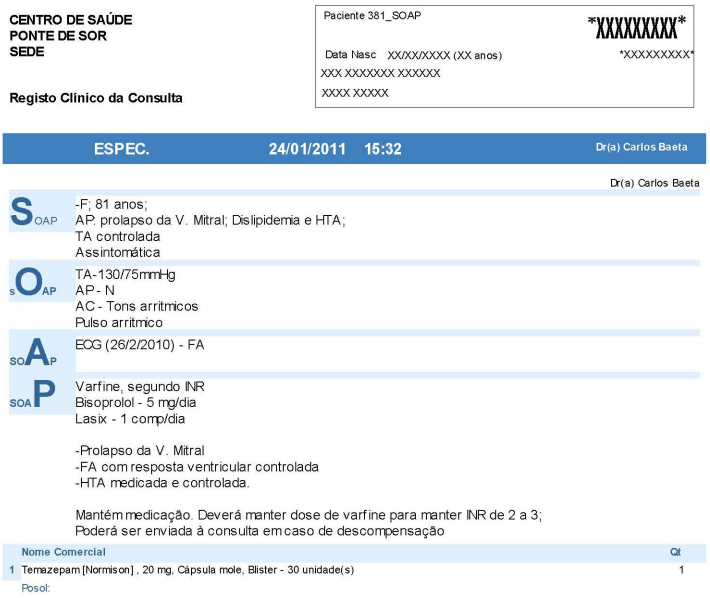
A set of sequential steps must be used to go from the pure text to the extracted CPR instance. Initially these tasks are done manually but after the initial proof of concept and tool customization they are automated. Those steps workflow can be configured declaratively using the software architecture shown in section 4. There are steps involved that consist of:

- PDF to raw text or to structured (XML) converting for adequate documents cleansing. For instance the graphical presentation of Vital Signs that are originally rendered in the respective report has to be deleted from the document for easier terms processing and the tables with values must be structured accordingly for the annotators to behave properly. Initially there is a proof of concept that involves manually cleaning the original reports
- Manual translation (that is indispensable for the translator tutoring as shown in 3.3) with the precise clinicians validation of their jargon adequately translated into English,
- Annotation using the Web interface of any of the services that we introduce in 3.5, either manually the interactive interfaces or automatically the Web Services available
- Filtering the concepts from the annotated text to insert in CPR instances

Given the array of available Web Services that can semantically annotate biomedical concepts in English in 3.5, we choosed to use an evolutionary approach for use of the BioPortal annotator [3]. We mean by evolutionary approach the fact that we first use the annotator after manual pre-processing and then a more automatic workflow.

3.3 The Multilingual problem

We can take advantage of the fact that we have to translate from jargon to English to customize the Google translator toolkit[6] with our own Translation Memories and Glossaries. Let us introduce some demonstrative examples taken from a sample document gently provided by Dr. Carlos Baeta and properly de-identified:



CENTRO DE SAÚDE
PONTE DE SOR
SEDE

Registo Clínico da Consulta

Paciente 381_SOAP *XXXXXXXXX*
Data Nasc XX/XX/XXXX (XX anos) *XXXXXXXXX*
XXX XXXXXXX XXXXXX
XXXX XXXXX

ESPEC. 24/01/2011 15:32 Dr(a) Carlos Baeta

Dr(a) Carlos Baeta

S -F; 81 anos;
AP: prolapso da V. Mitral; Dislipidemia e HTA;
TA controlada
Assintomática

O TA-130/75mmHg
AP - N
AC - Tons arritmicos
Pulso arritmico

A ECG (26/2/2010) - FA

P Varfine, segundo INR
Bisoprolol - 5 mg/dia
Lasix - 1 comp/dia

-Prolapso da V. Mitral
-FA com resposta ventricular controlada
-HTA medicada e controlada.

Mantém medicação. Deverá manter dose de varfine para manter INR de 2 a 3;
Poderá ser enviada à consulta em caso de descompensação

	Nome Comercial	Qt
1	Temazepam [Normison] , 20 mg, Cápsula mole, Blister - 30 unidade(s)	1

Posol:

Fig. 2. SOAP 381

We will, in the process of using the Google toolkit, create Translation Memories with the identified personal acronyms like:

- AP (Antecedentes Pessoais) into Personal History
- HTA (Hiper Tensão Arterial) into High Blood Pressure
- FA (Fibrilhação Auricular) into Atrial Fibrilation
- V. Mitral (Válvula Mitral) into Mitral Valve

Some which are acronyms that can be given the suitable translated concept like:

- ECG (Electro Cardio Grama) into Electro Cardio Gram

or those that are even English acronyms:

[6] https://translate.google.com/toolkit

– INR (International Normalized Ratio) into International Normalized Ratio

and some which are not really needed because the conveyed information is irrespective of what language is in like:

– CENTRO DE SAÚDE into HEALTH CENTER
– SEDE into MAIN OFFICE

Included in this sample are notorious some more complex problems that are not related to the translation itself but with some other problems like the time spanning of concepts like "1 comp/dia" which is adequately translated to "1 tablet per day" using the defined Translation Memory but has to be posteriorly well defined as time delimited occurring process this kind of problems and the suggested solutions are itemized ahead in 3.4.

3.4 SAM Corpus

In our particular case we face a shortage of structure of the reports extracted from SAM in order to be able to fill the suitable instances in CPR in a more systematic way. As an example we illustrate below that no complete demographic information is available in any of the used reports or that the problems enumerated in the Problem List don't have any kind of severity or progress information associated. Reports that are produced by the SAM system and have interest for our work are:

– List of Medical Problems

This report lists dated references to medical problems that were found to be of reporting interest at any particular moment in the patients live. It provides a history line of clinical problems without any relation to any procedures taken whatsoever. One of the interesting contributions of automatic acquiring this information into our ontology is the possibility to generate a problem oriented timeline into which all the further discovered clinical procedures can be related to. It lacks, however, any kind of data referring intensity or gravity of the problem or its evolution along the time frame from when it was declared active until the problem dismissal. So far, as what is of interest to our Ontological Realism approach, it can be seen as a possibility of reasoning by fact inferring and it renders then our proposal somewhat more interesting because that possibility can be easily demonstrated. The pre-processing step of this type of documents has to take into appropriate care the fact that concepts extracted can be found in a "feeder ontology" like MeSHPOR and thus have an ID from a standard vocabulary associated or not. From problem we can then extract possibilities that will allow us to identify if there is an associated etiologic agent, a pathological-disposition or a mere sign recording for instance and then give the possibility to refine the medical problem instance with further information that shall always agree to this pre-defined structure:

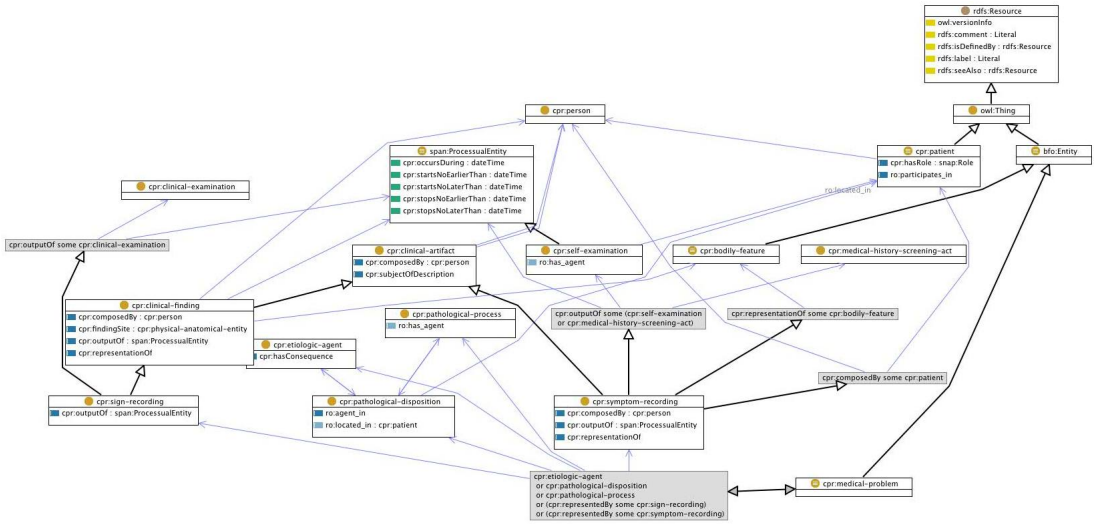


Fig. 3. CPR Medical Problems

This medical problems are, of course, an cpr:hipothesizedProblem of a cpr:clinical-diagnosis as we can see in the diagnoses view of the CPR ontology:

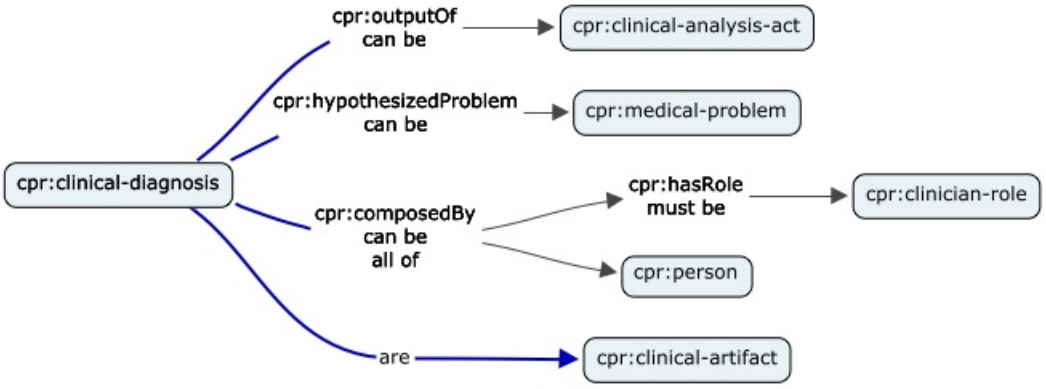


Fig. 4. CPR Diagnoses view

– SOAP Report

This report is the most informative of all available as it depicts a clinical encounter in a semi-structured way. As seen previously in the figure in section 3.3 we find sections that can be associated with

Symptoms, the subjective section S where we extract directly to cpr:symptom-recording.

Signs, the objective section O that are cpr:sign-recording that we take as generator for cpr:clinical-findings.

Actions, the analysis section A which are the cpr:clinical-investigation-act whose outputs can be cpr:clinical-artifact to investigate things that can be cpr:isConsequenceOf any of cpr:physiological-process or cpr:pathological-process

and finally

Plan, the plan section P where the therapeutic acts can be extracted with all the timing, posology and prescriptions registered in a particular clinical encounter.

– Different Areas Exams

This report has a summary of the different diagnostic complementary exams that a certain patient has been subjected to. It is divided into Laboratory Analysis, Pathological Anatomy, Common Exams and Imagiology. All these cpr:clinical-analysis-act contribute, or not, to a cpr:clinical-diagnosis that is a cpr:hypothesizedProblem of a cpr:medical-problem. This can be the workhorse of the automatic acquisition because the reports that are based in free-text can have origin from the different EHR modules and validated in advance rendering a high degree of certainty and even the possibility of pre-encoding according to some controlled vocabulary like ICD-9 or ulterior for example.

– Vital Signs Summary

The Vital Signs Report has to have the biggest amount of text cleansing of all because all the graphics have to be taken out and the tables have to be formatted as such for the annotators to understand them as tables and do the appropriate treatment that is take each line at once and create the specific instance in cpr:clinical-finding.

3.5 Services to annotate clinical concepts in free text

Apart from being able to provide "our own" Web Services for various tasks given the availability of downloading several types of terminologies like MeSH or SNOMED CT CORE and generally the UMLS Metathesaurus, currently there are a miriad of WS at reach that can be configured to be connected to our CP-ESB as service providers. Among those we think that are worth mentioned here the BIOPortal [7], OntoCAT[8] and UTS[9].

All these provide Web Services that offer specific tasks for Biomedicine terminology. Carefully chosen endpoints provide features that range from simple term lookups to complete semantic concept acquisition. Normally all these offerings are available at no cost upon registering and access granting.

[7] http://bioportal.bioontology.org
[8] http://www.ontocat.org
[9] https://uts.nlm.nih.gov/home.html

4 Software Architecture

To have an extensible architecture able to build upon and capable of entailing the current available tools we chose to use Java based tools. Namely building upon an Eclipse based modularization platform called OSGi[10]. The OSGi technology is a set of specifications that define a dynamic component system for Java. These specifications enable a development model where applications are (dynamically) composed of many different (reusable) components. The OSGi specifications enable components to hide their implementations from other components while communicating through services, which are objects that are specifically shared between components. This surprisingly simple model has far reaching effects for almost any aspect of the software development process.

Though components have been on the horizon for a long time, so far they failed to make good on their promises. OSGi is the first technology that actually succeeded with a component system that is solving many real problems in software development. Adopters of OSGi technology see significantly reduced complexity in almost all aspects of development. Code is easier to write and test, reuse is increased, build systems become significantly simpler, deployment is more manageable, bugs are detected early, and the runtime provides an enormous insight into what is running.

The OSGi technology was aimed to create a collaborative software environment. Here an application emerges from putting together different reusable components that had no a-priory knowledge of each other. The goal is to allow the functions to be added without requiring that the developers have intricate knowledge of each other and let the components be added independently.

One of the major architectural components that fosters the decoupling of the different components is a common rail where messaging can flow using a subscription model that enables the communication to be detached from any two particular services but instead be available on-request by one and served by another in a loosely coupled way. ESB[11] is a modular and component based architectural component. It assumes that services are generally autonomous and availability of a service at a certain moment of time cannot be guaranteed. Therefore messages need to be routed consequently through the message bus for buffering (message queuing to allow inspection and enhancement of content as well as filtering, correction and rerouting of message flow. In an enterprise architecture that makes use of an ESB, an application will communicate via the bus, which acts as the single message turntable between applications. That approach reduces the number of point-to-point connections between communicating applications. This, in turn, makes impact analysis for major software changes simpler and more straightforward. By reducing the number of points-of-contact from and to a particular application, it is easier to monitor for failure and misbehavior in highly complex systems and allows easier changing of components.

[10] www.osgi.org
[11] Enterprise Service Bus

It is an essential design concept of an ESB that every client directs all its requests through the ESB instead of passing it directly to a potential server. This indirection allows the ESB to monitor and log the traffic. The ESB can then intervene in message exchange and overwrite standard rules for service execution. The case of an intervention here is in the ability to filter and redirect invocations to the appropriate NLP task processors depending on the source being labeled with the kind of load it carries. For example, if an invocation carries the language labeled as PT it will invoke the MeSH concept translator before invoking the rest of the NLP processing pipeline. Another example may be workflow maintained in the ESB itself of the invocation of the CPO refiner before the CPO populator. The pipeline is maintained by configuration of the ESB and not hard-coded in any way. Buffering and delaying message in a staging area and automatically deliver it when the receiver is ready, monitor messages and services to be well-behaved, enforce compliance with dynamic processing and security policies, marshal service execution based on dynamic rules, prioritize, delay, and reschedule message delivery and service execution, write logs and raise exception alerts all are examples of the ESB workhorse functionality. Notably the REST kind of software architecture, that has in the World Wide Web its most prominent example, is suggested in our proposal as the correct way of implementing a SOA [12] that serves as the communication underlying structure of our system. REST interfaces are available for consuming for the generality of our needs as shown ahead. In the next figure we present the alignment of the process invocation to fill the points suggested in [4] for disease and diagnosis representation in our CPR ontology.

CP-ESB The advantages of using a Software component as important as the ESB in the current SOA world of application composition is beyond the scope of this work. An important source of information to get up-to-date might be the Wikipedia page: http://en.wikipedia.org/wiki/Enterprise_service_bus. In our case the applicability of technology to the Clinical Practice environment can be seen as:

[12] Service Oriented Architecture

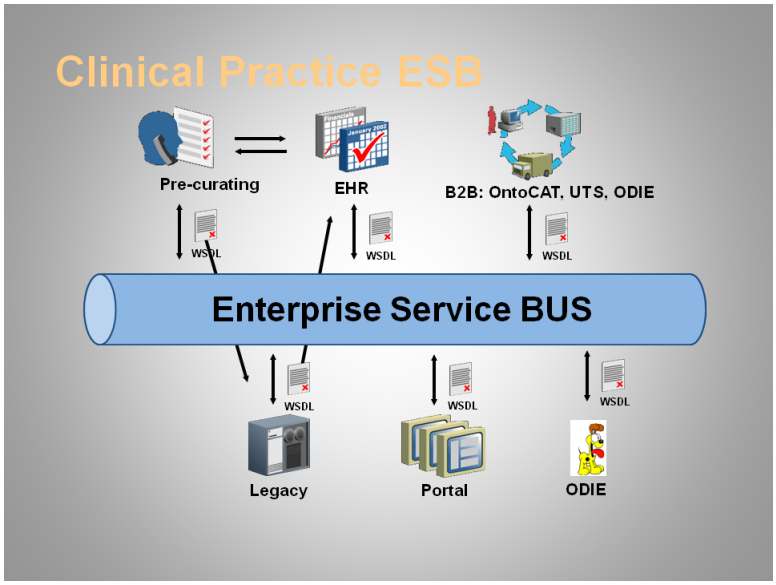


Fig. 5. Clinical Practice - ESB

Where the articulation of the providers and the Ontologies are well defined and are not handled point-to-point but always through the ESB routing and intercepting capabilities. All the core features are already implemented to enable plug-and-play ability for interchanged modules as stated above.

Connecting the dots Building over the suggested infrastructure the systems are rather composed as opposed to monolithicaly built and so manifest high capabilities of plug-and-play configuration allowing for interchangeable providers (as Web Services), Reference Ontologies (Feeders), and target ontologies. Having the foundations available with the right weapons provided one has to take a practical approach to the development of a target system using, in our case, the Java best-practices for pragmatic development that include a number of Patterns as in JEE [13] compiled in http://java.sun.com/blueprints/corej2eepatterns/Patterns/ or the pragmatic approaches developed in such successful projects as OSGi or Spring[14].

The flowchart that depicts graphically the acquisition from the source texts in Portuguese to the creation of the appropriate CPR instance is:

[13] Java Enterprise Edition
[14] http://www.springsource.com/

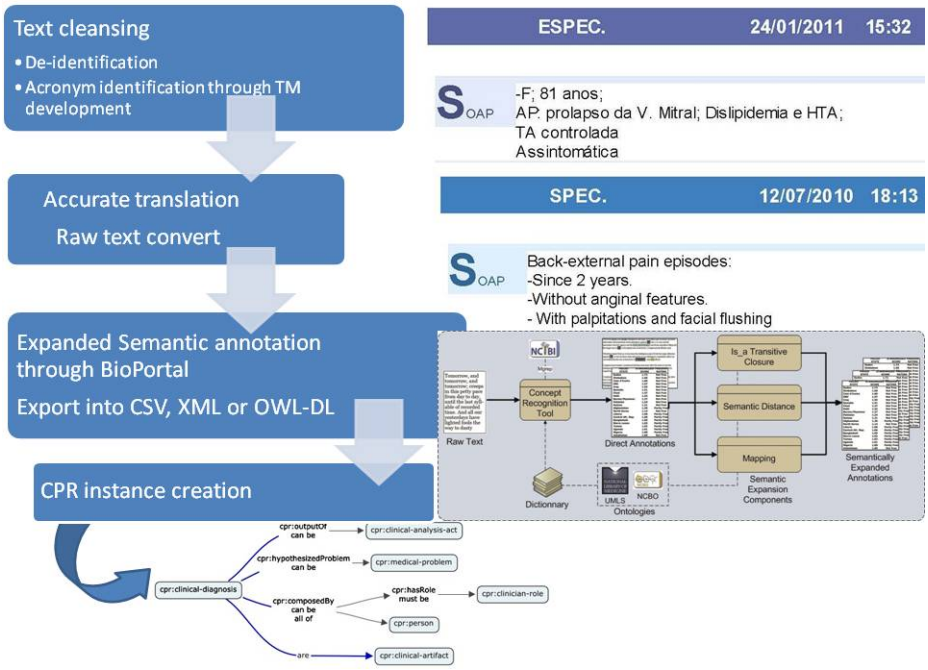


Fig. 6. Acquisition Flowchart

ODIE stands for "Ontology Development and Information Extraction". It is a software toolkit that uses ontologies to perform information extraction tasks from clinical documents and uses clinical documents to enhance existing ontologies. Both ODIE and BioPortal code document sets with ontologies or enrich existing ontologies with new concepts from the document set. They contain algorithms for Named Entity Recognition, Co-reference resolution, concept discovery, discourse reasoning and attribute value extraction. They allow development of reusable software leveraging existing NCBO tools and compatible with NCBO architecture. A downloadable version gives the possibility of developing local extensions to the algorithms provided in the base offering allowing, for instance, targeting different languages in the NLP tasks. The WS provided by BioPortal or OntoCAT can be locally extended and refined for all the sources are provided as one of the projects deliverables.

5 Conclusion

We presented a humble contribution to demonstrate the articulation needed of different software tools and medical knowledge to be able to fill a Clinical Practice Ontology with instances collected automatically from reports taken from a specific local EHR system.

Bibliography

[1] Ceusters, W., Smith, B., Kumar, A., Dhaen, C., 2004. Mistakes in medical ontologies: where do they come from and how can they be detected? Stud Health Technol Inform. 2.1

[2] Mendes, D., Rodrigues, I., 2011. A Semantic Web pragmatic approach to develop Clinical ontolologies, and thus Semantic Interoperability, based in HL7 v2.xml messaging. In: HCist 2011 - Proceedings of the International Workshop on Health and Social Care Information Systems and Technologies. Springer-Verlag - book of the CCIS series (Communications in Computer and Information Science). 3.1

[3] Noy, N. F., Shah, N. H., Whetzel, P. L., Dai, B., Dorf, M., Griffith, N., Jonquet, C., Rubin, D. L., Storey, M.-A., Chute, C. G., Musen, M. a., Jul. 2009. BioPortal: ontologies and integrated data resources at the click of a mouse. Nucleic acids research 37 (Web Server issue), W170–3.
URL http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2703982&tool=pmcentrez&rendertype=abstract 3.2

[4] Scheuermann, R. H., Ceusters, W., Smith, B., 2009. Toward an Ontological Treatment of Disease and Diagnosis. In: 2009 AMIA Summit on Translational Bioinformatics. San Francisco, CA, pp. 116–120. 2.1, 4

[5] Smith, B., Ashburner, M., Rosse, C., Bard, J., Bug, W., Ceusters, W., Goldberg, L. J., Eilbeck, K., Ireland, A., Mungall, C. J., Leontis, N., Rocca-Serra, P., Ruttenberg, A., Sansone, S.-A., Scheuermann, R. H., Shah, N., Whetzel, P. L., Lewis, S., Nov. 2007. The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration. Nature biotechnology 25 (11), 1251–5.
URL http://www.ncbi.nlm.nih.gov/pubmed/17989687 2.1

Well Formed Clinical Practice Ontology Selection

David Mendes; Irene Pimenta Rodrigues

Departamento de Informática da Universidade de Évora

{dmendes;ipr}@uevora.pt

Abstract. We show how carefull shall an ontology selection process be in the specifig sub-domain of healthcare practice. This Ontology shall be well suited to reason about the clinical practice for it has to be based upon current Semantic Web techniques. Namely reasoners over OWL DL[1] ontologies have to handle the choosed ontology in such a way that the representational capabilities go hand-in-hand with adequate computability. We present the choice of ontology with all the theoretical considerations that have to be taken and show why the CPR[2] ontology is the best suited for our enrichment/population endeavours.

1 Introduction

We will present in this paper the reasons and causes of choosing CPR Ontology as the basis for our Clinical Practice domain Knowledge Representation.

1.1 Motivation

Since the early years of our century a large body of research has been developed in the Biomedical domain of knowedge. Beggining in 2006, the work around an ontology to adequately represent the healthcare providing activities has been around with a first proposal in 2009 as the CPR ontology. The Semantic Web tools and techniques have come of age to be able to use an ontology about a specific scientific and/or professional domain as knowledge representation scaffolding enough to be able to reason automatically and semantically inter-operate in that domain so we present here the selection process of such an ontology in a timely manner.

1.2 Previous work done

We are at the very beginning of the first author PhD work development under tutoring of the second. So far, only studying about the subject of Health Information knowledge representation and the Semantic Web tooling to reason around it has been done. To further develop and demonstrate the applicability of our work we have to choose and/or develop or enhance so we have been developing a carefull selection for a significant amount of time.

[1] Web Ontology Language - Description Logic
[2] Computer Based Record Ontology

2 What Ontology to Populate

Deciding what Ontology to Populate to function as the KB[3] to our Semantic Web Reasoning efforts is by itself a daunting task.

- The medical practice we want to represent is a many faceted science that renders a complex domain with issues to be addressed as
  - Temporality
  - Location
  - Granularity
  - High ambiguity in free text terminology
  - Jargon plagued with acronyms and even personal nicknames
- The Ontology shall take in consideration several different "best-practices" to be highly usable and used as intended
  - Solid design foundations for proper Ontology alignment and interoperability

  Well formed ontologies are able to support a variety of secondary uses not anticipated when the ontology was originally conceived [8]. In the process of trying to figure out the availability of such an ontology suitable for our purposes we found that the simplest was to develop an architectural software foundation to deliver them according to the Ontology Realism principles enunciated in [6] and with the freedom to be extendable according to anyone's particular needs. The ontologies here introduced that are to be in accordance to the OBO Foundry principles and thus interoperable may be a subset of any system brought up from our proposal. We just try to bring together the latest Software Engineering principles to the Ontology Engineering findings introduced in the referred article. With the loose coupling availability, configurable service inter-mixing, we picked what we could spot has low-hanging fruit to incorporate in our systems rendering them sub-optimal but demonstrable of the validity of the concepts and easily extendable/tunable with better ontology support and finer Web Service provisioning. For the moment the more widely accepted reference terminologies in form of ontologies that can be Web Service accessed through OntoCAT[4] or ODIE[5][6] were used and all the major coding standards that are similarly available were choosed. Given the impracticalities of using the whole UMLS, the relations and groups in the Semantic Network as preserved and only the MeSH[7][8] tractable terms and appropriate CORE views of SNOMED CT[9] are reached.

[3] Knowledge Base
[4] http://www.ontocat.org/
[5] Ontology Development & Information Extraction
[6] http://www.bioontology.org/ODIE
[7] Medical Subject Headings
[8] http://www.nlm.nih.gov/mesh/
[9] Systematized Nomenclature of Medicine - Clinical Terms

- Direction towards CSI[10]

Our coordinated, consistent ontologies shall be the structure for the Shared Meaning that is the most important concept to achieve CSI. The layer of understanding that is to be reached among disparate systems shall lie in an abstraction layer above the specific intricacies of each Clinical System. Considerations about the use of initiatives and already existing deliverables like HL7 V3 CDA[11], GreenCDA[12], CDISC[13] or BRIDG[14] are seriously considered.

- Integration for Extension

When trying to extend a particular given ontology to make it fit a particular purpose some techniques have been presented in the Ontology Engineering field. Some can be traced back to the early years of our century like Guarino and Welty [2]. Some new approaches are currently under heavy development and attracting special interest, these are mainly revolving around treating the ontologies as Metadata themselves and being able to process them for integration, clustering, validation or various other objectives. Work appearing recently is not reviewed here but we find projects like OASIS: Ontology Mapping and Integration Framework [9] and EL-VIRA [3] worth mentioning.

- Ontological Realism

Bayegan [1] proposed a process ontology for Clinical Practice which incorporated family-care workflow processes, clinical activities, different participants, and interactions of participants with a patient-record system. It was particularly interesting because it defined the minimal number of clinical headings necessary in a clinical setting, and also because their model was compatible with HL7. A similar effort was carried out by Scheuermann et al. [5], where they mainly focused on the disease and diagnosis manifestations. There are also some top-level ontologies in the literature (e.g., BFO[15] , BIOTOP, etc.) which can be further customized and expanded. Inspired by all these initiatives, W3C proposed an OWL-DL Computer-based Patient Record (CPR) ontology[16], which is briefly described in the following section 2.1

2.1 CPR

Computer-based Patient Record (CPR) was defined by the Institute of Medicine (IOM) in 1997 as an electronic patient record that resides in a system specifically designed to support users by providing accessibility to complete and accurate data, alerts, reminders, clinical decision support systems, links to medical knowledge, and other aids. Mostly they have been generally implemented and known

[10] Computer Semantic Interoperability
[11] Clinical Document Architecture
[12] http://wiki.hl7.org/index.php?title=GreenCDA_Project
[13] Clinical Data Interchange Standards Consortium
[14] Biomedical Research Integrated Domain Group
[15] http://www.ifomis.org/bfo
[16] http://code.google.com/p/cprontology

as Electronic Health Records (EHR) during recent years. The CPR Ontology addresses the terminology requirements of a CPR and its recording contents. These are defined as uniform core data elements, standardized coding systems and formats, a common data dictionary and information on outcomes of care and functional status. The ontology defines a minimal set of terms. It provides principled, ontological commitment for the terms used in many of the health-care information terminology systems. CPR relies on the use of foundational ontologies and ontology engineering best practices namely the OBO Foundry principles adherence are a requisite in its formation. Lastly it's intended to be used as an upper ontology of clinical medicine such as the OGMS[17]. In order to achieve uniformity, it needs to have significant coverage which turns into a pyramid ontology paradigm: small, well organized top and wide idiosyncratic bottom as seen in Figure 1

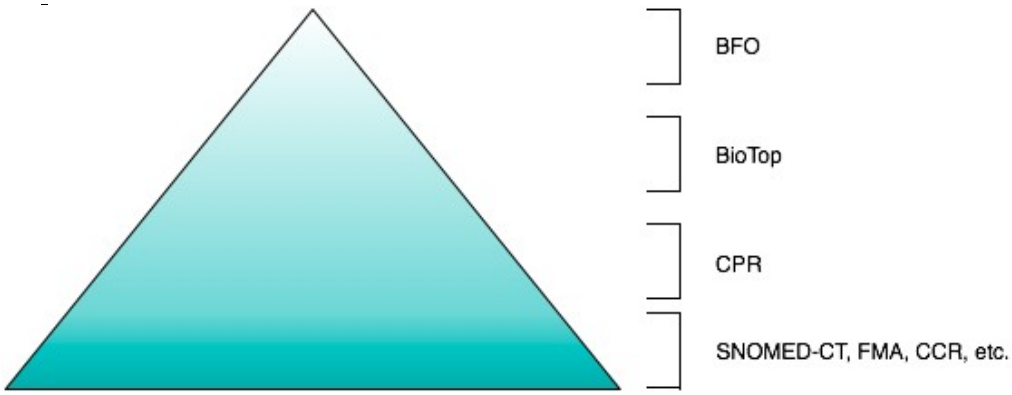


Fig. 1. CPR Pyramid

It adopts a cogent conceptual models that appeal to an ontological study of clinical medicine. High degree of care was taken into the development of CPR considering:

- The adoption of Ontological Realism has introduced by Barry et al. [7] according to the OBO Foundry principles
- Clear separation by definition of situations, findings and observables
- Differentiation among representational artifacts v.s. their referents
- Care act hierarchy and clinical workflow [1]
- Disease, diagnosis, etiology and the Disease Entity Model
- Disease, diagnosis, bodily features, etc. [5]
- Integrating anatomy, physiology, and pathology.

Use of realist ontology approach to the extent that distinctions are useful for real-world clinical informatics problems and validate against data and standard,

[17] Ontology for General Medical Science

controlled vocabularies namely SNOMED CT and FMA. There is a reasonable consensus around two reference ontologies that cover a substantial portion of clinical medicine: SNOMED-CT and the FMA. The location of equivalencies between classes and the extracted concepts from text is one of the major issues in our work so that the resulting populated ontology renders a realistic picture of the care process whose texts are the source of knowledge.

History and Motivation W3C first started to develop a Problem-Oriented Medical Record Ontology in 2006. The goal was to define a minimal set of health-care information terms while ontologically grounding HL7 RIM as a process model and using the criteria outlined in the traditional POMR structure W3C [11]. This led to the Web Ontology Language (OWL)-based ontology in November 2009, called the Computer-based Patient Record (CPR) ontology W3C [10]. Some parts of this ontology were taken from other top-level ontologies (e.g., BFO 1.1, BIOTOP, FMA, etc.) to ensure a sound and coherent means of necessary terminological representations required by an EHR. The surgical contributions to CPR has led into an ontology profoundly aligned with some basic "feeder ontologies" all of them according to OBO-Foundry principles and this alignment can de depicted as:

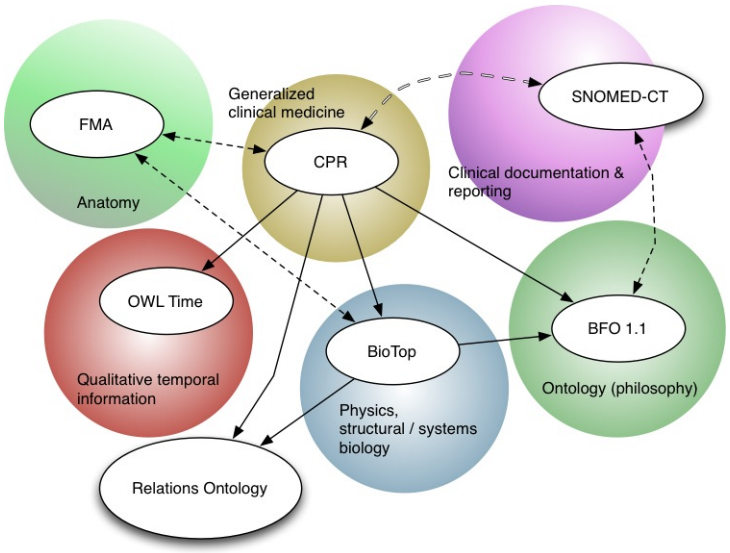


Fig. 2. CPR Ontology Alignment

Structure and Extensibility The main core concepts of this ontology are shown in the figure

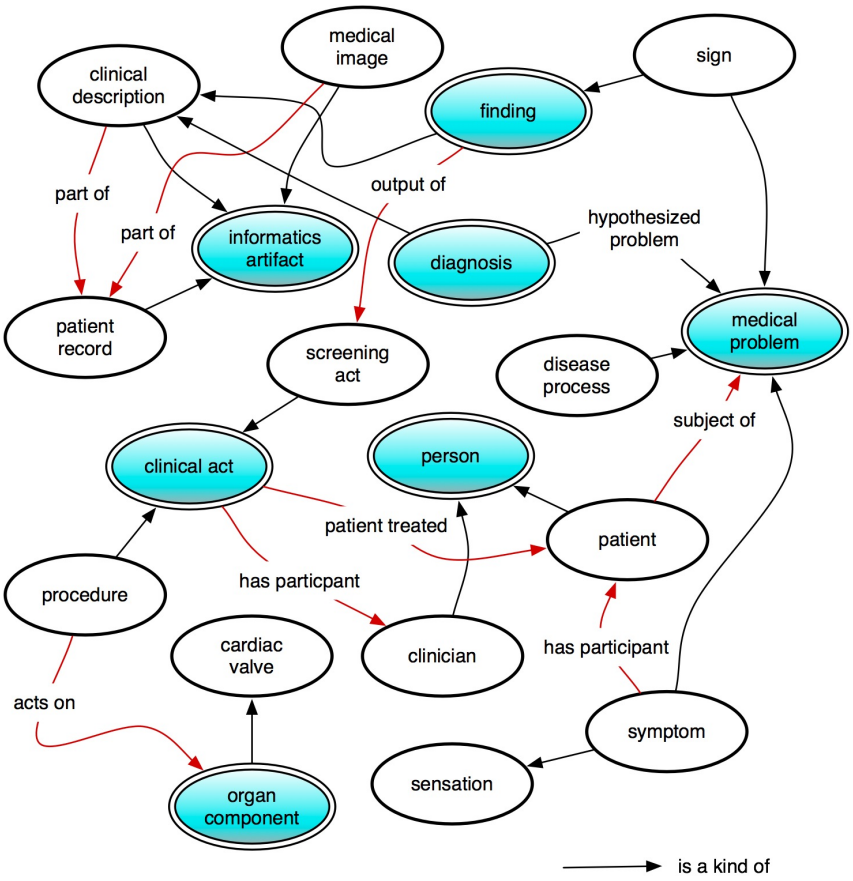


Fig. 3. Concepts of CPR ontology

2.2 Classes of CPR that are to be populated

We automatically extract information that will populate classes with creation of instances. The information available is far from what should be considered like a minimum to render a full clinical practice ontology filling. Some trade-offs must be made and we shall squeeze all the texts in order to get as most as possible according to the CPR classes pre-defined. In the figure depicting CPR structure in 2.1 the top-level concepts of the CPR archetypes are shaded and shown with double circles. These are described below:

Clinical Acts: The most important concept of CPR ontology is Clinical Acts, which is used to model various clinical tasks and activities and the information flow in these activities. This ontology used the process ontology of defining clinical processes as a workflow model proposed by Bayegan et al. Bayegan [1] for defining the minimum clinical headings that are important for

clinical communication and documentation. These clinical headings were put under the 'span:Process' class of BFO Ontology [45] to ensure proper classification of ocurrent and continuants data.There are four specializations of Clinical Acts: Clinical Administration Act, Clinical Investigation Act, Procedure, and Therapeutic Act. A Clinical Administration Act is defined as any administrative act which is not itself investigatory or therapeutic and is done for either the assessment or treatment (e.g., patient appointment). A Clinical Investigation Act is used to discover the status, causes and mechanisms of a patient's health condition and is further classified into four classes: Clinical Analysis Act (used to generate the clinical hypothesis based on the condition of disease, physical examination, lab results, etc.), Diagnostic Procedure (the process of assessing the diagnosis; includes both laboratory or radiological procedures), Laboratory Tests (the process of quantitative or qualitative test of a substance in laboratory), Screening Act (collecting data from different aspects (e.g., clinical examination, medical history, social history, family history, etc.) to identify problems). A Procedure is a type of act which is taken to improve the patient's condition. This concept is used in this ontology to incorporate both diagnostic and therapeutic procedures and is aligned with the definition of Procedure in HL7 RIM. Therapeutic acts are activities which are taken to improve or maintain the physical condition of a patient. This incorporates medical therapy (e.g., surgery), physical therapy (e.g., exercise), and psychological therapy (e.g., request to read an article that will improve the patient's psychological status).

Medical Problems: In this ontology, medical problems are defined as entities which incorporate the signs, symptoms and confirmed diseases of a patient. Signs are abnormalities interpreted by clinicians during physical examinations whereas symptoms are particular sensations reported by the patient themselves. The disease process has been defined as either pathological disease or etiological agents while re-using the ontological framework for disease and diagnosis proposed by Scheuermann et al. Scheuermann et al. [5].

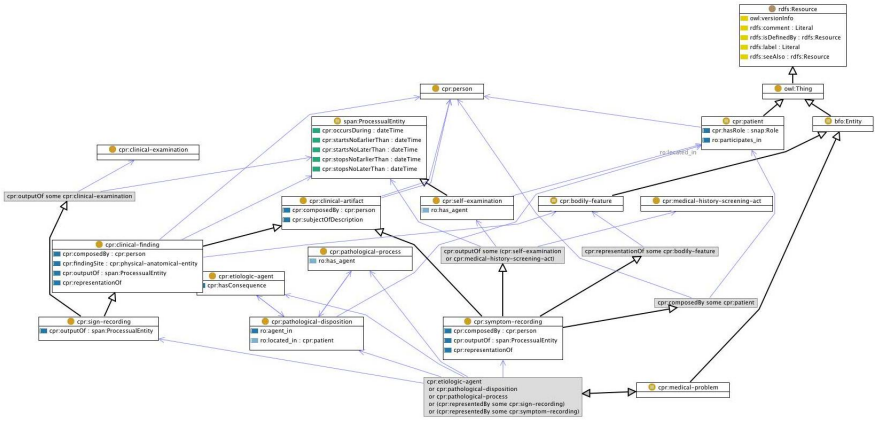


Fig. 4. CPR Medical Problems

This medical problems are, of course, an cpr:hipothesizedProblem of a cpr:clinical-diagnosis as we can see in the diagnosis view of the CPR ontology.

Findings: Findings are clinical examinations done by a clinical expert during an encounter to assess the condition of patient's body parts.

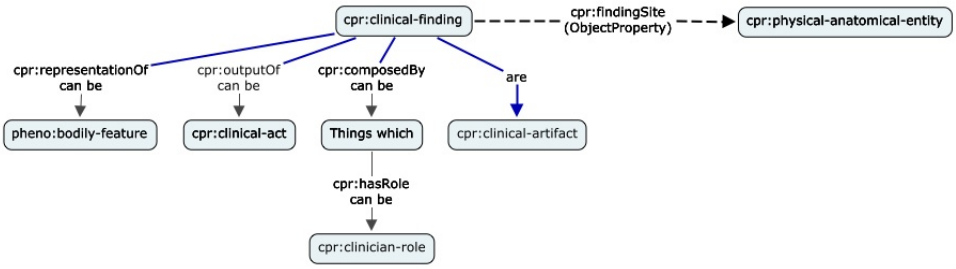


Fig. 5. Findings

Diagnosis: Diagnosis is not confirmed but hypothesized medical problem recorded during clinical analysis acts.

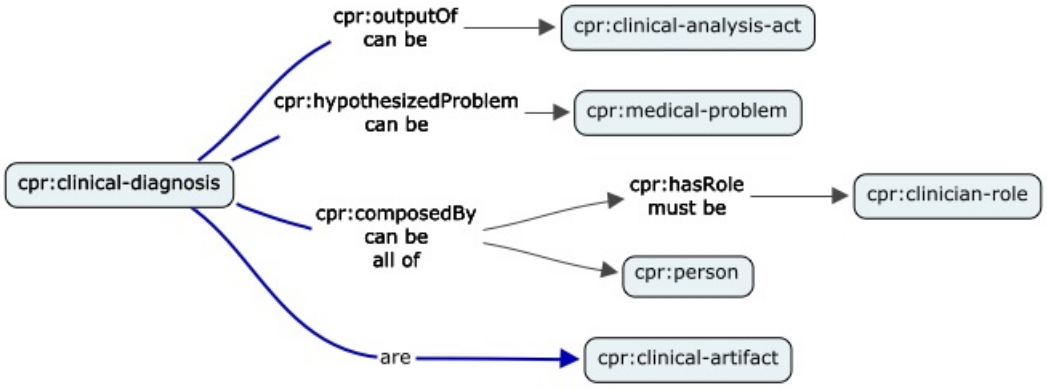


Fig. 6. CPR Diagnosis view

Informatics Artifacts: Informatics artifacts represent the pertinent information stored in an EHR. It includes all the clinical artifacts encountered in a patient, digital entities (e.g., diagnostic images), and other longitudinal information (e.g., clinical findings, symptoms). This concept is used to distinguish between the records of an action and the actual action itself.

Person: A person can be either the patient him- or herself or the clinically qualified person (e.g., nurse, general practitioner, etc.).

Organ Components: Organ components are the anatomical and pathological entities which take part in different clinical procedures and screening acts.

The CPR ontology is engineered in Protégé using OWL-DL language. Although it has all the necessary concepts an EHR should have, it lacks the properties of these concepts and the implementation of vocabulary binding in this ontology. To overcome this shortage we suggest the validation against the vocabularies that are translated partially to Portuguese and clinicians should be familiar with MeSHPOR.

2.3 CPR integrated with MeSHPOR

To define the properties of the concepts of this ontology a corresponding well defined and suitable vocabulary has to be adopted. Also, the vocabulary should be bound to this ontology so that the EHR concepts can use coded values where necessary like those that can be taken from the free text acquisition or from digging the EHR databases. We consider integrating for this purpose with the Portuguese localizations of MeSH[18] this version of the Medical Subject Headings is maintained and released annualy by the Latin-American and Caribbean Center on Health Sciences Information (Centro Latino-Americano e do Caribe

[18] http://www.nlm.nih.gov/research/umls/sourcereleasedocs/current/MSHPOR

de Informação em Ciências da Saúde). The current version contains 26142 Main Headings from wich 14902 designated synonyms can be extracted. This wealthy resource of Portuguese translated terms will allow us, using a simple Ontological Engineering technique, to bind terms to created instances in CPR ontology.

3 Conclusion

We presented a humble contribution to demonstrate the cautions required to select a clinical practice Ontology suitable to be enriched/populated with instances collected automatically from reports taken from EHR or other colectable sources of information.

Bibliography

[1] Bayegan, E., 2002. Knowledge representation for relevance ranking of patient-record contents in primary-care situations. Ph.D. thesis, Norwegian University of Science and Technology, Faculty of Information Technology, Mathematics and Electrical Engineering, papers III and IV "This material is presented to ensure timely dissemination of scholarly and technical work. Copyright and all rights therein are retained by authors or by other copyright holders. All persons copying this information are expected to adhere to the terms and constraints invoked by each author's copyright. In most cases, these works may not be reposted without the explicit permission of the copyright holder.".

[2] Guarino, N., Welty, C., 2000. Identity, unity, and individuality: Towards a formal toolkit for ontological analysis.

[3] Hoehndorf, R., Dumontier, M., Oellrich, A., Wimalaratne, S., Rebholz-Schuhmann, D., Schofield, P., Gkoutos, G. V., 2011. A common layer of interoperability for biomedical ontologies based on owl el. Bioinformatics 27 (7), 1001–1008.
URL http://bioinformatics.oxfordjournals.org/content/27/7/1001.abstract

[4] Noy, N. F., Shah, N. H., Whetzel, P. L., Dai, B., Dorf, M., Griffith, N., Jonquet, C., Rubin, D. L., Storey, M.-A., Chute, C. G., Musen, M. a., Jul. 2009. BioPortal: ontologies and integrated data resources at the click of a mouse. Nucleic acids research 37 (Web Server issue), W170–3.
URL http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2703982

[5] Scheuermann, R. H., Ceusters, W., Smith, B., 2009. Toward an Ontological Treatment of Disease and Diagnosis. In: 2009 AMIA Summit on Translational Bioinformatics. San Francisco, CA, pp. 116–120.

[6] Smith, B., Ceusters, W., Nov. 2010. Ontological realism: A methodology for coordinated evolution of scientific ontologies. Applied ontology 5 (3-4), 139–188.
URL http://dx.doi.org/10.3233/AO-2010-0079

[7] Smith, B., Kumar, A., Ceusters, W., Rosse, C., Jan. 2005. On carcinomas and other pathological entities. Comparative and functional genomics 6 (7-8), 379–87.
URL http://www.pubmedcentral.nih.gov/

[8] Smith, B., Scheuermann, R. H., Jan. 2011. Ontologies for clinical and translational research: Introduction. Journal of biomedical informatics 44, 3–7.
URL http://www.ncbi.nlm.nih.gov/pubmed/21241822

[9] Song, G., Qian, Y., Liu, Y., Zhang, K., 2006. Oasis: A mapping and integration framework for biomedical ontologies. In: Proceedings of the 19th IEEE Symposium on Computer-Based Medical Systems. IEEE Computer Society, Washington, DC, USA, pp. 611–616.
URL http://dl.acm.org/citation.cfm?id=1152999.1153011

[10] W3C, 2009. Compter-based patient record ontology.
URL http://code.google.com/p/cprontology
[11] W3C, 2009. Hcls pomrontology.
URL http://www.w3.org/wiki/HCLS/POMROntology

OLAP em âmbito hospitalar: Transformação de dados de enfermagem para análise multidimensional

João Silva and José Saias

m5672@alunos.uevora.pt, jsaias@di.uevora.pt
Mestrado em Engenharia Informática, Universidade de Évora

Resumo O desenvolvimento a que assistimos nos dias que correm levanos a ter que tomar decisões cada vez mais sustentadas e correctas. Tal encaminha-nos para uma procura incessante de mais e melhor informação.
O uso de bases de dados e a constante angariação de dados subjacente para salvaguardar e optimizar o funcionamento das organizações de diferentes áreas vieram trazer um grande avanço na perspectiva da colecta de dados que posteriormente se tornam numa grande fonte de conhecimentos.
Os Data Warehouses (DW), aliados a sistemas de Online Analytical Processing (OLAP), estão contidos no rol de soluções disponíveis para análise de dados e são uma grande mais valia para os analistas que vêem o seu trabalho bastante facilitado.
Este artigo visa descrever um pouco, estas duas tecnologias e mostrar como elas se complementam, levando ao desenvolvimento de uma ferramenta incluída na temática de Business Intelligence.

1 Introdução

O facto de estarmos bem informados sobre uma determinada área do nosso interesse sempre foi vital para o desenrolar do nosso comportamento e para qualquer tomada de decisão da nossa parte.

O problema que se põe é o facto de nem sempre a informação que procuramos estar disponível de forma clara. Isto leva-nos a procurar soluções sofisticadas de forma a conseguir descobrir essa informação, contida no meio de uma quantidade infindável de dados armazenados.

Quando falamos ao nível organizacional essas informações tomam proporções bastante maiores, uma vez que nas organizações qualquer mudança com vista a melhorar, pode trazer bastantes vantagens tanto ao nível funcional como principalmente ao nível lucrativo.

Durante muito tempo estas tomadas de decisão foram orientadas exclusivamente pelo saber administrativo, sem necessidade de se recorrer a bases de dados organizadas.

Nos dias que correm o uso das bases de dados e a constante angariação de dados durante o funcionamento das organizações trouxe-nos a possibilidade

de armazenar uma quantidade enorme de informação. Desta feita existe um conceito designado por *Business Intelligence* que podemos traduzir facilmente por Inteligência nos Negócios.

Existem várias técnicas ou metodologias englobadas no conceito de BI, onde todas elas têm o intuito comum de fornecer conhecimentos importantes.

Podemos então definir o conceito de BI um conjunto de componentes e processos, que juntos permitem a angariação dos dados provenientes de diversas fontes, organizando-os, processando-os e armazenando-os de forma correcta para que sejam apresentados ao utilizador final, de modo a facilitar o processo de tomada de decisões[9].

Neste contexto os sistemas de OLAP constituem uma das soluções mais utilizadas por parte das organizações, pois estas fornecem ao utilizador final a possibilidade de navegação pelos dados de forma bastante intuitiva, sendo possível a criação de tabelas, gráficos, entre outras possibilidades.

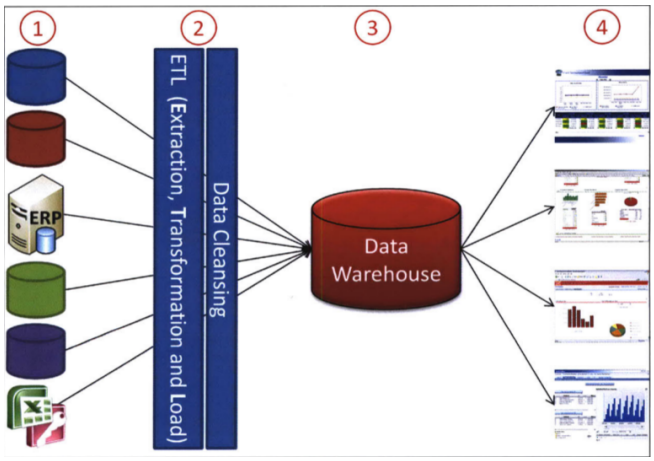


Figura 1. Diversos passos para o desenvolvimento de uma solução de BI[9]

O bom funcionamento destas ferramentas depende muito de uma boa implementação do repositório onde os dados analisados estão armazenados. Estes repositórios designam-se por Data Warehouses e, como podemos observar na figura 1, estes constituem a parte central e uma das partes mais importantes para o bom funcionamento deste tipo de ferramentas.

2 Data Warehouse

Ao longo dos anos, têm surgido diversas formas de definir um DW. Em termos teóricos e de acordo com W. H. Inmon, o seu principal arquitecto, "a data

warehouse is a **subject-oriented**, **integrated**, **time-variant**, and **nonvolatile** collection os data in support of management's decision making process"[4]. Por meio destes quatro termos é possível descrever as características de um DW, isto é:

- Subject-oriented: um DW é desenvolvido e organizado, de modo a satisfazer as necessidades de análise de uma organização, relativamente a um ou mais aspectos chave. Relativamente ao trabalho realizado e descrito neste artigo, um dos aspectos chave foi a Taxa de prevalência relativa aos diversos diagnósticos, presentes nos registos de enfermagem.
- Integrated: um DW é por norma desenvolvido, utilizando diversas fontes de dados externas, como bases de dados relacionais, folhas de cálculo, entre outras. Como tal, alguns problemas de consistência dos dados necessitam de resolução.
- Nonvolatile: um DW apenas permite o carregamento dos e a leitura dos mesmos. Operações de modificação e de remoção não são permitidas.
- Time-variant: os dados são armazenados de modo a fornecerem uma perspectiva histórica dos dados.

A necessidade do uso deste tipo especial de armazenamento surge, devido ao facto de as bases de dados convencionais estarem mais optimizadas para gerir um grande número de transacções e um constante fluxo de dados com a preocupação de manter a consistência dos dados. Como tal, não estão preparadas para em tempo útil processar consultas complexas, que são efectuadas por sistemas de análise como é o caso do OLAP[3].

Em suma estes são descritos como uma arquitectura que obedece a determinadas especificações técnicas, sendo formado através da integração de diversas fontes de dados, para o suporte de sistemas de análise com o intuito de optimizar e apoiar as organizações no processo de tomada de decisões.

2.1 Modelo Multidimentional

Para proporcionarem uma boa fluidez os DW baseiam-se no modelo multidimensional, também conhecido por *Cubo de dados*.

Através da observação da figura 2 podemos constatar a existência de **dimensões** que fornecem o contexto ao utilizador sobre o qual se deseja analisar as **medidas** existentes. Estas medidas correspondem aos valores quantificáveis que são usados para analisar as relações entre as dimensões[3]. Um exemplo possível e referente ao trabalho realizado, é a análise da medida *Taxa de Prevalência* num determinado contexto, fornecido pelas dimensões *Tempo*, *Geográfica* e *Diagnóstico*.

De referir que dentro de cada dimensão, existem *hierarquias*, estruturadas de modo a definirem vários níveis de granularidade, que pode ser maior ou menor conforme se sobe ou desce, respectivamente na hierarquia[5]. Dentro da dimensão tempo, uma hierarquia possível é a sequência constituída por anos, seguida de meses e por último dias. Sendo que o nível anos é o que possui menor granularidade e por sua vez o nível dias o que possui maior. Por último, cada nível de

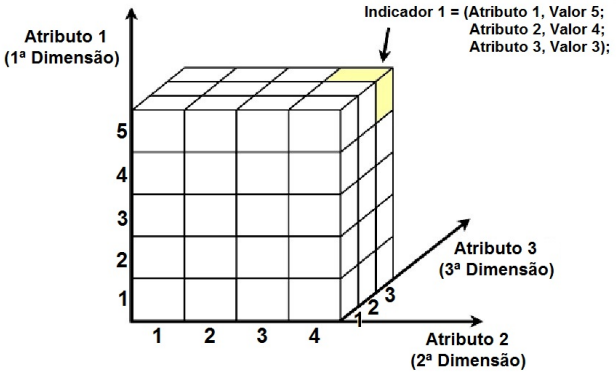


Figura 2. Cubo de dados

uma hierarquia possui os seus *membros*, o que permite filtrar os dados dentro de uma dimensão, caso seja necessário encontrar uma situação mais concreta.

2.2 Esquema em Estrela e Esquema em Floco de Neve

Estes dois esquema representam duas maneiras de implementar o modelo multidimensional em bases de dados relacionais.

O esquema em estrela é o principal esquema utilizado e, ao mesmo tempo, o mais comum[1]. Contém uma tabela central normalmente designada por *tabela de factos*, ou seja, é a tabela que contém os factos que podem corresponder às medidas ou que possibilitam o seu cálculo, e por sua vez é a que possui maior quantidade de dados. À volta desta estão tabelas mais pequenas que representam as tabelas das dimensões.

A diferença entre o anterior e o esquema em floco de neve é a normalização das tabelas das dimensões, pois em vez de cada dimensão ser constituída por uma só tabela, estas estão divididas em várias.

Com esta normalização, algumas hierarquias passam a ser explícitas[10]. Esta vem trazer benefícios ao nível do espaço utilizado pelas tabelas das dimensões, sendo necessário menos espaço de armazenamento, diminuindo também a redundância destas tabelas. Contudo, dado que as tabelas estão separadas, são precisos mais *joins* aquando da execução de consultas, o que pode afectar bastante a performance.

2.3 Extracção, Transformação e Carregamento (ETL)

Posteriormente à estruturação do **DW**, inicia-se o processo de **ETL**. Este processo é por norma o mais difícil e moroso quando se trata do desenvolvimento deste tipo de soluções[9]. O principal objectivo é a angariação e a transferência dos dados de diversas fontes de uma organização para o **DW**, ou seja, juntar os

dados, por norma heterogéneos, para uma representação homogénea que permita processos de análise eficazes e eficientes.

Por vezes, deve ser realizado o processo de *Limpeza dos Dados* antes que estes sejam extraídos, transformados e finalmente carregados para o DW. Tal deve-se ao facto de os dados nem sempre se apresentarem de forma correcta ou completa, criando problemas de consistência.

3 Servidores OLAP

Entre o DW e as ferramentas de análise, existe ainda uma camada bastante importante, designada por *Servidor OLAP*, dado que OLAP funciona através do conceito multi-utilizador cliente/servidor[6].

Os dois servidores OLAP mais utilizados são o **M**ultidimensional **OLAP** e o **R**elational **OLAP** e a grande diferença entre eles é a forma como os dados são armazenados.

Multidimentional OLAP Como o nome indica no MOLAP os dados são armazenados de forma multidimensional, isto é, em estruturas multidimensionais do tipo array. O seu funcionamento baseia-se no calculo prévio das agregações de diversas combinações das dimensões existentes, sendo estas armazenadas nas estruturas mencionadas anteriormente[11].

Este tipo de armazenamento não é muito apropriado aquando da existência de um grande número de dimensões dado que ao efectuar o pré-calculo de todas as combinações possíveis, o tempo necessário para efectuar esta operação pode tornar-se bastante elevado.

Relational OLAP Ao contrário do anterior, este tipo de servidor utiliza a própria base de dados relacional como forma de armazenamento. Como tal, o seu papel é servir de intermediário entre o servidor relacional, onde estão guardados os dados, e o cliente, estendendo as capacidades destes servidores. Deste modo estes passam a suportar as consultas multidimensionais, características das ferramentas OLAP[2].

Por norma quando se trata de um conjunto baixo de dimensões a diferença de performance para o MOLAP não é significativa e no caso de haver um aumento no número destas ou no volume de dados, então o ROLAP ganha vantagem pois como não necessita de efectuar o cálculo prévio das agregações, o tempo de resposta a este cenário é bem mais rápido[11].

4 Operações OLAP

Recorrendo à organização dos dados de forma multidimensional e aliado ao facto das dimensões serem hierárquicas, as ferramentas de OLAP permitem uma grande flexibilidade de navegação por entre os dados.

As operações mais importantes fornecidas por estas ferramentas são as seguintes:

- **Drill-Up** ou **Roll-Up** - permite subir na hierarquia de uma dimensão ou mesmo de remove-la. Por exemplo, passar de uma vista por cidades para a vista por distritos, subindo na hierarquia da dimensão localização;
- **Drill-Down** - permite efectuar oposto da operação anterior;
- **Slice e Dice** - permitem efectuar cortes na visualização dos dados. Por exemplo, podemos querer visualizar os dados relativos apenas ao primeiro trimestre do ano 2010, o que corresponde a efectuar um *slice* na dimensão tempo;
- **Drill-Through** - permite observar a fonte dos dados que deram origem a uma determinada agregação.

Embora estas sejam as principais funcionalidades normalmente presentes neste tipo de ferramentas, existem outras, como a capacidade criar gráficos, exportar os dados observados para outros formatos como PDF ou Excel, etc.

5 Trabalho Realizado

Com base nos conhecimentos adquiridos que foram, de uma forma resumida, descritos nas secções anteriores deste artigo, falta então descrever um pouco o trabalho realizado. Através deste foi desenvolvida uma solução de OLAP de modo a permitir a análise de medidas, relativas a registos de enfermagem. Como tal, foi necessário efectuar todo o processo de construção de um DW, criando o repositório multidimensional, efectuando o processo de ETL necessário.

Para a realização do mesmo foram utilizadas ferramentas open source disponíveis para a comunidade que, à excepção do sistema de gestão de bases de dados MySQL[7], são pertencentes à empresa Pentaho[8], responsável pelo desenvolvimento de tecnologias direccionadas para a temática de BI.

5.1 Repositório Multidimentional

O primeiro passo foi desenvolver o repositório multidimensional, ou seja, a base de dados que constitui o DW.

Na fase inicial utilizou-se um esquema em estrela para desenvolver a base de dados, tal como mostra a figura 3. Através da observação deste podemos perceber quais as dimensões utilizadas, dentro destas quais os campos que permitem formar as hierarquias de cada uma e por último quais os factos que permitem o cálculo da medida referente a esta tabela de factos. A medida *taxa de prevalência* é posteriormente calculada através do campo *codigo_pacientes* que constituem os factos desta tabela.

Dado que este trabalho não se referia a apenas uma medida, ao esquema da figura 3, foram adicionadas mais duas tabelas de factos em que estas possuem dimensões exclusivas de cada uma e outras partilhadas entre si.

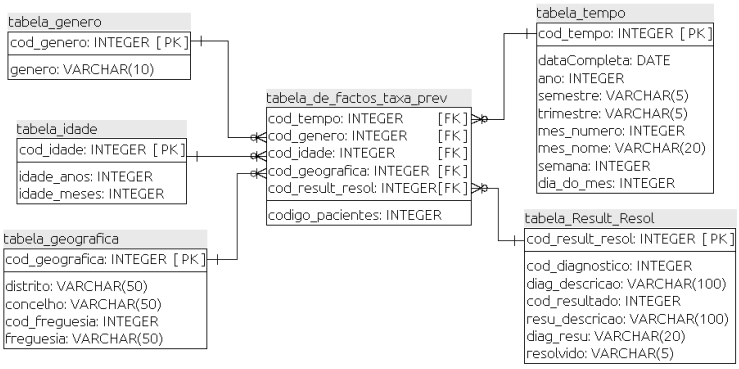


Figura 3. Esquema em estrela

5.2 Processo de ETL

Depois de construído o repositório, o passo seguinte foi extrair os dados da única fonte existente, uma base de dados relacional pertencente a um sistema proprietário de suporte ao processo de registos de enfermagem. De seguida efectuar as transformações necessárias e por último carrega-los para o DW. Para tal foi utilizada uma ferramenta designada por Pentaho Data Integration, a qual permitiu efectuar todo este processo.

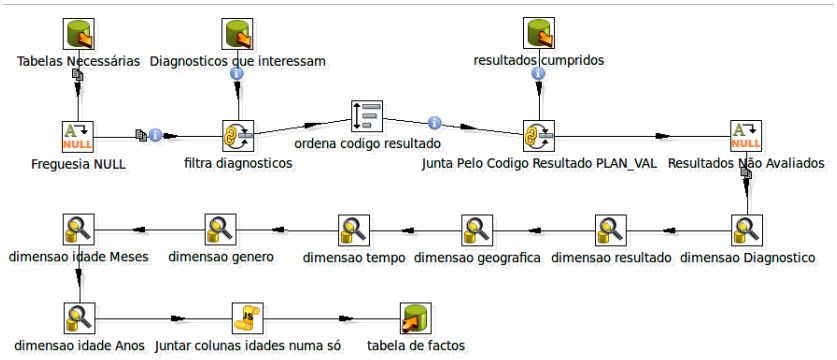


Figura 4. Transformação para a tabela de factos referente à medida Taxa de Prevalência

Como podemos observar pela figura 4, à medida que vão sendo feitas as consultas à base de dados fonte, de modo a extrair a informação necessária, vão sendo efectuadas as transformações nos dados extraídos através dos passos seguintes.

Como as tabelas de factos possuem as chaves primárias de cada dimensão, foi necessário encontrar as diferentes chaves correspondentes para cada facto antes de ser efectuado o carregamento, Como tal, foi necessário efectuar um processo semelhante de carregamento para todas as dimensões, de modo a ser possível fornecer o contexto a cada facto.

5.3 OLAP

Para desenvolver a fase final do trabalho, foi utilizado um servidor OLAP baseado na arquitectura ROLAP, designado por Pentaho BI Server. Aliado a este foi utilizado também uma ferramenta de visualização designado por STPivot, responsável pelo fornecimento das operações OLAP anteriormente referidas.

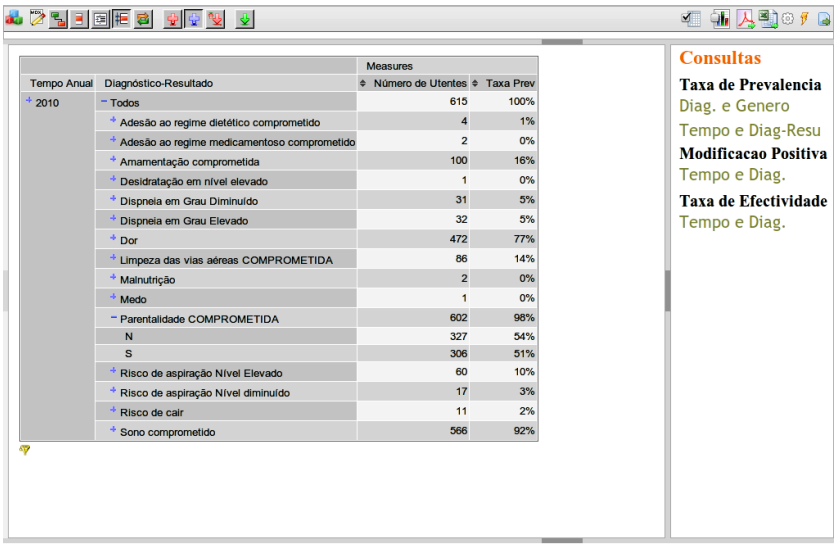


Figura 5. Consulta efectuada através do STPivot

A figura 5 mostra o aspecto final da ferramenta de OLAP e representa a consulta da taxa de prevalência por tempo, neste caso referente ao ano de 2010 e por diagnóstico. Nesta dimensão foi efectuado o *drill-down*, mostrando todos os diagnósticos existentes no ano de 2010 e por sua vez um segundo drill-down, mostrando os resultados resolvidos e não resolvidos para o diagnóstico *Parentalidade Comprometida* no mesmo ano.

De referir que as consultas são efectuadas numa linguagem multidimensional desenvolvida exclusivamente para os sistemas OLAP, designada por MDX. Como tal foi necessário criar um esquema através da ferramenta Schema Workbench, responsável por fornecer ao servidor ROLAP a informação necessária de modo a que este consiga efectuar a tradução das consultas MDX para código SQL referente à base de dados que constitui o DW.

6 Conclusão

Todas estas tecnologias tanto de armazenamento, como de análise, entre outras, têm vindo a evoluir bastante e de modo síncrono, dado que se trata de tecnologias complementares. Embora sejam os sistemas OLAP que permitem a interacção com os dados, não nos podemos esquecer que sem os Data Warehouses, estas se tornariam inúteis, pois uma a boa estruturação dos dados é fundamental para o bom funcionamento destas ferramentas de análise.

OLAP tem vindo a ganhar um papel cada vez mais importante na vida das organizações, não só na área dos negócios mas também noutras, como a da saúde, onde a tomada de decisões é bastante importante de modo a ganhar uma maior eficiência e rapidez nos serviços.

Como exemplo disso está o trabalho realizado onde o objectivo principal foi desenvolver um sistema de OLAP para a área da saúde mais propriamente para análise de registos de enfermagem.

Referências

1. Gauree Bhole. Building a data mart using star schema, 2010.
2. S. Chaudhuri and U. Dayal. An overview of data warehousing and olap technology. 1997.
3. J. Han and M. Kamber. Data mining: Concepts and techniques, 2000.
4. W. H. Inmon. *Building the Data Warehouse*. QED Technical Publishing Group, Wellesley, Massachusetts, 1992.
5. E. Malinowski and E. Zimányi. Hierarchies in a multidimensional model: From conceptual modeling to logical representation. *Data & Knowledge Engineering*, 2006.
6. Fred Curtis Moulton. Olap and olap server definitions. `http://www.moulton.com/olap/olap.glossary.html`. Acedido em Outubro de 2011.
7. Mysql. Mysql: The world's most popular open source database. `http://www.mysql.com/`.
8. Pentaho. Pentaho open source business intelligence. `http://www.pentaho.com`.
9. Eumir P. Reyes. A systems thinking approach to business intelligence solutions based on cloud computing, 2010.
10. Yin Jenny Tam. Datacube: Its implementation and application in olap mining, 1998.
11. Per Westerlund. Business intelligence: Multidimensional data analysis, 2008.

Applying Problem Based Learning for improving Human-tech competencies in Computer Engineering students: a research proposal

Nuno Valero Ribeiro[1], Joaquim Filipe[1], and Carlos Pampulim Caldeira[2]

[1] Escola Superior de Tecnologia de Setúbal
Instituto Politécnico de Setúbal
2910-761 Setúbal, Portugal
{nuno.ribeiro, joaquim.filipe}@estsetubal.ips.pt,
[2] Departamento de Informática
Universidade de Évora
7000 Évora, Portugal
ccaldeira@di.uevora.pt

Abstract. This paper summarizes the background theory of the likewise entitled research project. The project aims to give a contribution to software programming quality improving "Human-tech" competencies in Computer Engineering students as a means to prevent, or at least avoid in a great extend, the rate of unsuccessful software implementation projects. We are specially interested in researching what Human Factors competencies must be profiled in Computing Curricula outcomes that may contribute to better prepare students as "Human-tech" experts. We will apply Problem Based Learning educational method for delivering those competencies to students. We believe it is possible to do better than what has been done, to have a better degree of adequacy between the Human user and the software used for his/her activity. All background theory that support the axiomatic principles of this research project are explained in the first section. Then the project is briefly outlined as well as its plan, expected outcomes and contribution, in the following sections.

1 Introduction and Background

1.1 Introduction

In this paper we demonstrate that there is a general need for a change in the way most software applications are built. Generally, software applications, such as isolated programs or complex Information Systems, are designed with a technocratic orientation in mind. Moreover, the driving forces are mainly those that come from the managerial owners. Software Engineers learn to deal with complexities of the technological tools they use in order to program software, but lack considering the Human-factors that may jeopardize the innovative process

of introducing that software in an organization. Therefore we believe that improving Human-tech competencies in Software Engineering students profile of learned outcomes is a necessity and would prevent the problem of inadequacy of software implementation when in use in an organizational context.

1.2 Information Systems

Summary:

1. IS's: general considerations and examples
2. Could *prevention* be a better strategy?

Current Status Implementation of Information Systems (IS's) (computer supported) in organizations do not always correspond the set of initial expectations. The experience, as a computer practitioner and as a computer engineer, has led us to frequently hear complains about IS's. Analyzing these complains, looking for the origin of these bad matches and their possible causes, one gets the traditional explanations: over estimated expectations, bad design plans, organizational resistance to changes, poor implementation, deficient planning of resources, lack of qualification, etc. But, digging more deeply, the following statements turn out to be the common link valid in all these cases:

> 1. Information Systems (computer supported) are typically considered as the "Holy Grail" that will solve all problems of an organization.

Instead, they become the problem itself. Time and money are consumed, attempts to fix the problems are expensively made, workers despair trying to use it, client complains tend to be excused blaming the computer, managers ask for solutions and developers try their best.

> 2. End users are neglected (specially) in the planning and development phases of Information Systems (computer supported).

The lack of consideration and participation of end users during the planning and development phases of the project typically leads to complains in the implementation phase. If IS end users are not consulted, their particular needs (vs. organizational) are not predicted or considered, and their jobs will not improve with the implementation of the IS. Structural coupling between IS's and its end users should always be considered as a *must have* requirement. Information Systems are typically designed with the goal of improving organizational efficiency in mind and neglect what can be the end user needs to operate properly with its support. Naturally, deceptions may occur.

Examples Different aspects of the problems described above are illustrated in the following examples taken from real situations:

1. The "CBC" case (2006): CBC is a large enterprise whose business is assuring quality demands in several fields. CBC has experts whose main job is to visit different companies, geographically dispersed, and write a report as the result of their visit. CBC experts began complaining when started to use the IS for the task previously hand written. They complain having to spend more time now, and effort, dealing with the CBC's IS than practicing their expertise competencies. This was not what enterprise managers wanted when implemented the IS, nor what experts aspire to. Statements like: "the IS is inadequate to our job practices" or "the IS is a terrible headache" are generally heard coming from the expert. Similarly, this situation occurs in other organizational environments, like, for example, in health care where nurses or doctors spend more time operating the computer than with attending the patient, as they should, since they were trained for it.
2. The "STB" case (2008): STB is a large public organization that has an IS for supporting the normal business processes. This example refers the process of requesting a room reservation. Its quite simple: the user just has to fulfill an IS data form about the event, hour, day, etc., clicks on a button and waits for the answer. The answer appears as a warning in the IS front end after a while. Nothing new. The problem occurred when the IS user was at the room door, with the confirmed reservation ticket at hand, which was produced via the IS, and soon realized that the room was occupied by someone else for another event. He was totaly embarrassed, and deeply ashamed. He had asked VIP's to come, speakers, who had previously confirmed their agendas and carefully prepared the conference presentations, plus friends and colleagues he had invited to attend the conference. Worst of all, there was no other room available. He asked for responsibilities since he did every action correctly and the reservation was validated. It just happened that the room was occupied by an event that was not registered in the IS. Why? Because the occupying event was considered so "important" (and needed so many rooms) that fulfilling IS forms was just a waste of time. The work load of the administrative functionaries in the weeks before was so big that instructions were given for not to worry about "infocratic" work and it was assumed that everyone knew about the big event. Well, unfortunately, not everyone knew.
3. The "SIT" case (2009): SIT is an IS that supports the ordinary actions of several participants in a distritbuted organizational process for a public large organization. The documents involved in the process, and stored in the IS, are classified and have high security requisites like personal electronic identification cards for user authentication. Thus, the IS has an uncommon authentication mechanism. One of the key participants in a business process had a health problem, leading to a health license and a big period of time absente in the process. A replacement had to be done in the process. The problem was to deal with the security constrains of the IS. How to replace the user that fulfils a function within a process since it has already began?

The IS administrator started by assigning the new user to the process, but soon realized that a new user could not access previously stored documents owned and registered by the replaced user. He tried to override the previously assigned user for these documents and assigned them to the new one. But, someone reminded him, he would put in stake the true authorship of the those documents. Actually, he did try, but the IS didn't allow it. Well, after spending hours dealing with the problem, phoning experts, etc., a decision was made. It was allowed for the new user to access the IS with the authentication privileges of the former user, handling him the former authentication "personal" card. This required a personal trust relationship among both of them: the former user had to lend the authentication card to the new one, and, the new one didn't mind "signing" his documents as another person. An odd situation considering the security requisites of the present IS.

1.3 Preventing instead of Correcting

Technicians strive for designing and implementing IS's capable of supporting all organizational business processes. The major drive force is a political and managerial desire to control and operate everything within the organization. This is mainly due to the globalization and economical trends that lead organizations to become off-shored and to have *on the fly* data for managerial purposes (see [12] for complete explanation of the subject). Adequation of IS's to this need is thus imposed from the top management as a necessity and does not come from the bottom. Users are generally forced to adapt their usual procedures to become compatible with the planed procedures implemented via the IS.

On the other hand, life is full of unexpected and uncontrolled events and organizations are living cells. As technicians improve and add functionalities to the IS, in order to support new business processes or to solve unexpected problems, the IS starts to endlessly grow and soon becomes an impossible-to-use-and-manage Babel.

This is due to the typically adopted approach to deal with the problem. The solution is more focused on a *detection and correction* strategy. Thus, a never ending story of maintenance and post-development efforts are generated. Focusing the solution on a *prevention* strategy, or, at least, *avoiding* mismatches at a greater extent, although initially more expensive, could be rewarding at the end. We believe that situations like those exemplified above could be prevented in the designing phase of technology if end users were more taken into account.

In order to explore solutions based on *prevention* or *avoiding* strategies one must first break the *sine qua non* conditions of the unsuccessful implementations of Information Systems. Development approaches in this engineering area still ignore its human and social connections, and implications, as pointed out in [3].

1.4 Software Engineering particularities

Summary:

1. Software Engineering vs. other engineering disciplines
   (a) The immaturity of Software Engineering field;
   (b) Base Sciences that support Software Engineering;
2. Axiomatic set of principles for Software Engineering

When comparing Software Engineering with other traditional engineering fields (like Mechanical Engineering, Electrical Engineering, etc.) we must agree that: 1. Software Engineering is more recent and, therefore, still in an "adolescent phase", full of experimental practices; and 2. Software Engineering is intrinsically different in the sense that traditional engineering base sciences do not apply to the field (like cinematic physics or electromagnetic physics).

1. The immaturity of Software Engineering field.
   In traditional engineering fields those who plan and design the technological artifact (for instance, a house, a bridge, an electronic motherboard or electrical plant) become specialized in that area, i.e. *planning and design*. Their concern is different from those who are concerned with building the artifact. Moreover, those who check if the technological artifact is being built accordingly to the initial planned design, or those who test it, may represent another group of specialized engineers.
   Software Engineering does not yet benefit from these separation of concerns and specialization, at least, at the extent that traditional engineering areas benefit. Software Engineering specialized offices for validating initial design plans, or for testing software developed by third parties, are to much expensive and the procedure is not applied. At least, the procedure is not practiced as it is in other traditional engineering fields. Most of times, in software business, not even the owner knows what he really wants with the implementation of an IS or software technology in his organization.
   Traditional engineering fields have become specialized through time. Design plans and methodologies for building their technological artifacts (like electronic maps, architectural plants, etc.) became consensually accepted and normalized. Generally those include accurate calculus and drawings, or procedures. This is due not only because of its maturity but also because those engineering areas are an application of exact sciences like physics.
2. Base Sciences that support Software Engineering;
   Base sciences are those from which the *application of* leads to a particular engineering field. The materials "crafted" by software engineers are not tangible. As the name implies they're soft! They're more related with information, people and organization, then with physical phenomenons studied in Electromagnetism, Electronics, Mechanics, etc. This fact makes Software Engineering a different "kind" of engineering field. We must take into account theories coming from human sciences like social, business, organizational sciences.

As we said, Software Engineering deals with people, organization and information: ways of storing, processing, changing, organizing, retrieving, reaching, displaying, arranging information, and so on. And, Software Engineering doesn't

have, at the moment, a commonly accepted conceptual way for planning or designing IS's or software. Thus, we may conclude that:

> Software Engineering has, in its foundation, exact sciences as well as human sciences.

Nevertheless, human sciences theories have been experimented in methodological approaches for designing IS's. For example, DEMO [4] methodology — Design & Engineering Methodology for Organizations — adopts a basic pattern of coordination as the building block for designing all business processes. This pattern is based on Habermas's *Theory of Communicative Action* [9] borrowed from linguistics. Or the MEASUR Organizational Semiotics based methods covered in chapter 4 of [13].

This methodologies are focused on a basic principle of mapping the organization's business processes in an *as-is* fashion for designing the IS at as a starting point. They are necessary as a starting point. Afterwards, the organization evolves for itself, demanding changes to the IS.

Other patterns of conversation do exist in organizations, and therefore, not all aspects of interaction among persons in an organization are supported via these approaches (we do not say they should be, just stating the fact). For example, a person may spontaneously interact with another person emitting a judgement about some piece of work. This pattern is not captured, unless if making part of a well-known business process. Although it is not part of the organization's regular business procedures, it might influence the organization evolution. It may even begin an innovative process in the organization and thus a future correction and development in the IS that supports it.

1.5 The User and the Organization

An organization is fundamentally a social environment as it is recognized by Dietz's in [5]. Since organizations are "living cells", narrowing the design of IS's to a particular or momentary view of the organization could represent a limitation to its own evolution. Structural coupling between the IS (that supports the work being developed in the context of an organizational activity), and its users, should be as extensively achieved as possible. In other words, as a driver may adapt the car seat to his personal physical stature, the user of an Information System should be given the possibility of adapting it to his functional roles and usage within the organizational context.

This does not typically happen nowadays: IS's have the same generic front-end and functionalities for all users, regardless of their particular functional needs in the organization. The access to the IS functionalities is not designed taking into account the user as an individual person, with particular needs and roles within the organization. The design is centered on a generic mechanical point of view of the organization, where the user is taken as a *piece* of the organization's complex mechanisms. The vision contained in the (prevention) strategy for a solution is taken from Vicente in [15]:

> Its necessary to tailor the design of technology to people, rather than, pushing people to adapt and decipher technology.

As Robert Brigs explains in [1] *a good theory is a* model *of* cause-and-effect *to explain some phenomenon of interest. Every technology presumes a cause-and-effect. Every technology is built to improve some outcome.* Therefore, the question is: what outcome do we which to improve? The outcome we which to improve with Information Systems or software introduction in organizations will set the phenomenon of interest for our research. The expected outcome to improve with technology introduction in organizations must thus be clear.

1.6 What are computers for?

Since Information Systems are deployed with the support of computers nowadays, this leads to the question: what is the purpose of implementing an IS supported by computers in a organization? That, in turn, leads to the fundamental question: *What are computers for?*[3]

The answer to this question is not an obvious one although it may seem easy to give an answer since computers are widely used by many people and in many contexts nowadays. We have run the experience of asking the question to different classes of Software Engineering students and the fact is that, among them, we have collected a set of totally different answers to the question [see collection of data *1]. Thus we may say that there is no (common) understanding about the purpose of computers.

The user of technology should have a clear idea of the purpose that it has. This applies not only to the final user, in strict sense, but, and even more seriously, to software engineers, and computer programmers, and owners, since their responsibility is bigger when building, developing and owning IS's.

In the introductory chapter of the book *Organized Activity and its Support by Computer* [10], Holt discusses and formulates that

> Computers are for reducing the effort of carrying out organized activity.

assuming a broad sense of the terms "computers" — any computational device either connected or not in networks — and "organized activity" — a human universal that exists wherever and whenever people exist.

Furthermore, the author also points out:

> The practical side of the question "What are computers for?" is the question "How should computers be programmed?".

[3] This issue was brought to us by Anatol Holt during is stay as invited professor in Technology School of Setúbal, Portugal, with whom we had the opportunity to have long and challenging discussions about computers, most of them based on his book *Organized Activity and its Support by Computer* [10].

Thus, the more clearly we understand "*What are computers for?*" the more efficiently we obtain the expected outcome of using computers. Particularly, if we are builders of Information Systems, or programmers, we should have the necessary tools for facilitating this perception.

Still, many programming projects are developed by teams of programmers who have to deal with, and understand, complex human and organizational problems without the necessary tools, and owners who will never understand the programs that are written and operated on their behalf.

Once computers are meant for helping people carrying out an Organized Activity (OA for short), the effort we spend performing an OA, when supported by computers, should decrease. Otherwise computers wouldn't be fulfilling their purpose. Since operating computers also requires human effort, there should be a tradeoff, and an ideal point, between the effort spent when using computers to support an OA vs. the benefits or outcomes achieved with the use of computers. And thus, this may bring a conflict of interests between the organizational goal when implementing an IS and what the end-user needs.

1.7 The Mechanistic view of the world

1. IS's inflexibility to changes

Typically, software engineers have a mechanical view of situations. They are trained in programming. This implies pre-defining possible routes of programm execution. Their "natural" tendency is to build programmes that map the analyzed business processes, and its flow of execution, into the architecture of the IS in an "As-Is" fashion (read [2] for example). Exceptions to the foreseen flow of execution and generally prohibited. They do not think about how to cope with future changes or exceptions to the plan.

Attempts made in this direction become hard to maintain. For example, the approach in which TOA (Theory of Organized Activity) [11] is based, to map the organization, assumes that an user of an IS is a tuple [person+function]. Either persons or functions are constantly changing in organizations. Plus, the external environment also imposes changes to organizations (legislation, clients, natural phenomenons, etc.). Changes become difficult to implement in an IS designed in this fashion. Unexpected events are difficult to handle. We have adopted TOA in the development of an IS [14]. Soon it became difficult to manage. It constantly demanded updates to the functions that were fulfilled by the persons of the organization. We realized that the only possible solution was to ask the employees themselves to adapt their profile in the system. This implies a more "As-the-user-needs" fashion, or approach, to design than the "As-is" fashion. In the initial plan, a responsible settled the employees profile according to the function(s) they were fulfilling in the organization at that moment. We were thinking only about the user as someone performing a function and neglecting the user as an individual person capable of deciding for himself. We were adopting only a mechanical point of view of the organization and neglected the humanist point of view.

Detecting and correcting the IS to cope with changes requires big post-development efforts. The construction of an IS capable of coping with the pace of constant changes in the organization is a challenge.

> Since the major vehicle of change in organizations is the human being, centering design decisions in the human person is mandatory to achieve flexibility.

Software engineers must take into account that organizations are "living cells" that evolve for themselves. Thus, IS's should be flexible enough to allow constant implementation of changes in their construct. This approach contrasts with the "box" fashion of selling software or "package" approaches. IS's should be though and conceived considering a more (not completely planned) "Lego" approach. The user should choose his front-end interface from a menu of implemented functionalities.

Letting the user shape his front-end and set-up the IS's functionalities that he needs is likely to be the most suitable way of coping with the pace of changes that occur constantly in organizations. IS support should be tailored and shaped for that particular user but this may only be achieved if we give the possibility of choice to the user. If this becomes a reality, the "structural coupling", or degree of fitness, between the outcome achieved through the support of the activity by means of an IS and the user needs will increase. A person, although taken as a user of an IS that is fulfilling a function in the organization, has interests, needs and will of his one.

> To achieve a greater degree of fitness, when designing an IS, we must take into account the organizational goals as well as the user personal interests.

1.8 Tailoring Technology for People

In the book *The Human Factor: revolutionizing the way we live with technology* [15], Kim Vicente points out that scientific knowledge has been divided into two big groups: human sciences and technical sciences. Human scientists, when they look at the world, they focus primarily on people. Technical scientists have adopted a mechanistic view: they look primarily on the hardware or software. He defends that design of technology must take into account the characteristics or needs of people who will be using it because, people and technology interact and exist, side-by-side, in the real world. We must resist adopting a partial vision of the real world for orientation in the process of designing technology.

Human-tech Oriented Design Vicente suggest the compound word "Human-tech" to remind us that people and technology are both important aspects of the system. "Human" comes first to remind us that we should start by identifying our human and societal needs, not by glorifying some fancy widget in isolation;

"tech" is a means, not an end in itself, so it comes second. He proposes five human aspects that may guide the technology designer in his work. These are: physical; psychological; team; social; organizational; political. In each one of them, human sciences have a lot of possible contribution for helping design decisions. It is not a rigid separation of concerns, may suffer adaptations. If we consider the human characteristics that are relevant to the specific design problem we want to solve, everything becomes much, much simpler.

Vicente states that adopting a context-specific, problem-driven approach narrows down the amount and kind of knowledge that we need to consider to find an effective design solution. The adopted solution must respect the physical, psychological, etc., characteristics of the people that will be using it in order to build an harmonious relationship between people and technology.

Human-tech oriented design may serve as a guide for the desired "structural coupling" between technology and its users.

1.9 Problem Based Learning

In order to train programmers for the design of software solutions based in context-specific, problem-driven approaches, one must chose the most probably correct methodology and test if it is effective. Applying PBL (Problem Based Learning) seems to be an adequate educational method for delivering "Human-tech" competencies to programmers. Accordingly to the UCPBL Centre for Problem Based Learning located in Aalborg [7], this method is adequate for changing challenges in educational engineering schools who want to develop a more student centered solutions focused in the learning process and on the development of new competencies such as "Human-tech" in computer engineering students.

The pedagogical principles and historical background of PBL may be found in a special edition about PBL and ICT (Information and Communication Technology) [6]. Like the approach brought by the introduction of this educational method in the McMaster medical school of Canada [8] we believe that, similarly, "Human-tech" should be learned in practice focusing on the Human user of computers and on his/her requirements. By systematically analyzing the user problems, students will formulate questions, search for information lacking to problems, select their learning goals. The objective is to train the student to act and think as a Human-tech expert. The experience obtained by the student is expected to become meaningfully in his future as a professional.

The model integrates a number of pedagogical principles: problem-orientation, inter-disciplinarian, participant control, exemplary project, teamwork, and action learning. However, the model is practiced in various ways adapting it to local conditions, subject matters, skills of students and supervisors, etc. Thus, adaptation of the model to local conditions and "Human-tech" competencies to be learned is required.

2 Research Project Description

2.1 The Problem

To many software applications nowadays have a bad "structural coupling" with the persons who are using them. According to Vicente (previously cited), this is essentially due to the lack of "Human-tech" competencies in those who design and develop technology. As Vicente points out: "when the wizards of technology design their gadgets many consideration of human aspects are neglected, or even, not considered at all."

Programming is not an easy task and programmers are heavily trained in technical skills. Generally, computer engineering degrees do not have special courses for qualifying their students in "human" competencies. Those competencies are typically said to be transversally taught and are typically neglected comparing with technical specialized competencies. There is a necessity to improve "Human-tech" competences/skils deliverance to Computer Engineering students as a means to improve software production quality.

2.2 Research Question

What "Human-tech" competencies and skills are needed for Computer Engineering undergraduate students in order to improve "structural coupling" between persons and computers. What results are obtained when applying PBL educational method for improving those "Human-tech" competencies in Computer Engineering undergraduate students?

Sub-questions What is the "Human-tech" profile of competencies for a computer engineering undergraduate student?

How to evaluate if a student have learned those competencies?

Does PBL proves to be an adequate method for delivering "Human-tech" Competencies in Computer Engineering degrees?

What are the specific characteristics od the adopted model of PBL in the tested scenarios?

2.3 Problem Significance

If we have better "Human-tech" programmers we should have better adequation of software design and development to organizations and users. More efficient employment of software programmers resources. More efficient return of investments in software development projects.

3 Research Plan

3.1 Research Work Plan

Start: OUT 2012 ... End: OUT 2015

tasks and outcomes

1. Literature study and field study
   (a) Actual profile of Learning Outcomes in Computer Engineering Curriculum
   (b) Desired situation for Human-tech Oriented Design of software
   (c) Formulate hypothesis comparing both
2. Field research for construction of learning outcomes evaluation
3. Proposal of a PBL based model for enhancing Human-tech learning outcomes
4. Case studies
   (a) Apply and test the model in real situations
   (b) Analysis of results
   (c) Processing of the outcomes
5. Reporting

3.2 Research Outcome

The expected outcomes of the project will consist on the following:

1. a list of Human-tech Computer Engineering learning outcomes;
2. a PBL based model for learning Human-tech competencies;
3. a method for evaluating Human-tech learning outcomes;
4. an assessment of the model provided through its application on Computer Engineering curricula;

Several reports will be produced, as indicated in the work plan, some of which are expected to be published in international conferences and workshops.

3.3 Validation Process

The validation of the outcomes of this project are not possible in laboratory due to the essential nature of a learning environment that cannot be artificially reproduced. Thus, a possible validation may be through applying the model in a set of case studies, i.e. Computer Engineering curricula, and making an assessment, confronting the results obtained with reality reported.

3.4 Research Value

Are those outcomes worth the effort? Whose work might be more effective? Whose life might improve?

To many times although, the "invisible" costs of a unsuccessful software piece, or IS implementation, are neglected, specially for the human user side of the coin. We which to give a contribution to the improvement of the rate of successful software implementation, focusing specially in its adequation to the (Human) user and organization. We think its be possible to prevent, or, at least minimize or avoid, the rate software implementation mistakes that, sometimes, jeopardize the success of technology innovation in organizations. We believe it is possible to do better than what as been done, to have a better degree of adequacy, to avoid unsuccessful implementation of software.

References

1. Briggs, R.O.: On theory-driven design and deployment of collaboration systems. International Journal of Human-Computer Studies 64(7), pp. 573–582 (July 2006)
2. Castela, N., Tribolet, J.: AS-IS Continuous Representation in Organizational Engineering. In: ICEIS 2008: Proceedings of the Tenth International Conference On Enterprise Information Systems. vol. I, pp. 371–374. INSTICC (2008)
3. Cordeiro, J., Filipe, J., Liu, K.: NOMIS - A Human Centred Modelling Approach of Information Systems. In: Proceedings of the 4th International Workshop on Enterprise Systems and Technology. Athens, Greece (2010)
4. Dietz, J.L.: Enterprise Ontology: Theory and Methodology. Springer-Verlag Berlin (2006)
5. Dietz, J.L.: The Deep Structure of Business Processes. Communications of the ACM 49(5), pp. 59–64 (October 2006)
6. Dirckinck-Holmfeld, L.: Innovation of Problem Based Learning through ICT: Linking Local and Global Experiences. International Journal of Education and Development using Information and Communication Technology 5(1), pp. 3–12 (2006)
7. Enemark, S., Kolmos, A., Moesby, E.: Global network on engineering education research and expertise in PBL. Samlignsnummer för enstaka enskilt utgivna arbeteb (2006)
8. de Graaff, E., Kolmos, A., et al.: Problem Based Learning. final report of the Special Interest Group (SIG) B5, TREE (Teaching and Research in Engineering in Europe) Thematic Network of SOCRATES/Erasmus programme of the European Commission (August 2007), URL, http://www.unifi.it/tree/dl/oc/b5.pdf, accessed October 2011
9. Habermas, J.: The Theory of Communiative Action: Reason and Rationalization of Society. Polity Press, Cambridge (1984)
10. Holt, A.W.: Organized Activity and Its Support by Computer. Kluwer Academic Publishers (1997)
11. Holt, A.W.: The "Organized Activity" Foundation for Business Processes and Their Management. In: Business Process Management, Models, Techniques, and Empirical Studies. pp. 66–82. Springer-Verlag, London, UK (2000), http://portal.acm.org/citation.cfm?id=647778.757179
12. Levy, F., Murnane, R.J.: The New Division of Labor: How Computers are Creating the Next Job Market. Princeton University Press (2004)
13. Liu, K.: Semiotics In Information Systems Engineering. Cambridge University Press (2000)
14. Ribeiro, N.V., Delgado, J., Rolo, J., Lourenço, R.: Application of Theory of Organized Activity to a Real Case Study. In: Cases and Projects in Business Informatics. pp. 167–181. Logos Verlag Berlin, Berlin, DE (2006)
15. Vicente, K.: The Human Factor: revolutionizing the way we live with technology. Vintage Canada (2003)

RECICLAGEM DE IMPRESSORAS NO ENSINO DE COMPUTAÇÃO FÍSICA

S. Shahidian[1], R.P.Serralheiro[2], J.M.Serrano[3] e R.M.Machado[4]

1 ICAAM, Universidade de Évora, Apt.94, Núcleo de Mitra, shakib@uevora.pt
2 ICAAM, Universidade de Évora, Apt.94, Núcleo de Mitra, ricardo@uevora.pt
3 ICAAM, Universidade de Évora, Apt.94, Núcleo de Mitra, jmrs@uevora.pt
4 ICAAM, Universidade de Évora, Apt.94, Núcleo de Mitra, rmam@uevora.pt

RESUMO
Anualmente um número elevado de impressoras é descartado, sendo a sua maioria depositada em aterros sanitários e uma pequena parte triturada e derretida para a recuperação dos componentes metálicos. Na Universidade de Évora, desde 2007 tem-se desenvolvido uma nova abordagem: as impressoras em fim de vida são desmontadas e as peças usadas nas aulas de automação e computação física, servindo não só para proteger o meio ambiente, como também para reduzir o custo anual de funcionamento das aulas de laboratório. Uma das impressoras mais amplamente disponíveis é a Hp série 600 que foi fabricada entre 1994 e 2000. Estas impressoras, uma vez desmontadas, podem fornecer vários componentes úteis para as aulas de electrónica, tais como: 2 motores de passos, um motor 12V completo com engrenagem e carrinho, um termómetro de precisão LM35DZ, um relé de contacto, um interruptor de infra-vermelho, duas LDRs e um cabo LPT de alta qualidade que pode ser usado em projectos de automação a partir da porta LPT do PC. Os quatro anos de experiência demonstram que os alunos estão dispostos a desmontar o equipamento e cuidadosamente extrair os componentes. Com as ferramentas adequadas, dois alunos podem desmontar uma impressora e remover as peças em menos de 15 minutos. Este artigo apresenta os componentes individuais que podem ser removidos e fornece exemplos de circuitos e usos para as peças em aulas de electrónica e computação física.

Palavras chave: reciclagem de impressoras, desmantelamento de impressoras, ensino de electrónica

INTRODUÇÃO

O volume de lixo electrónico está a aumentar rapidamente. Em 2010, mais de 125 milhões de impressoras foram vendidas globalmente, dos quais mais de 52 milhões foram produzidas pela HP [1]. Isto representa um aumento de 12% sobre volumes de 2009, e este número deve continuar a aumentar. Assumindo um peso médio de 5 kg por unidade, esse número equivale a um total de 750 000 ton. de lixo electrónico que precisa de ser reciclado anualmente. O processo de reciclagem convencional consiste em triturar o equipamento até se obter um triturado fino, e depois separar os materiais usando as suas propriedades físicas e químicas. Os produtos finais deste processo de reciclagem são metais ferrosos, metais misturados, plásticos mistos e material indiferenciado. Em seguida, cada lote de materiais é vendido para processamento posterior.

Os metais constituem 57 por cento do peso total de produtos obtidos pela trituração de sucata electrónica [2], pelo que muitos processos de reciclagem se concentram principalmente na recuperação de metais e metais preciosos. Os metais utilizados são principalmente o ferro, ferro fundido, aço inoxidável e outras ligas de aço, alumínio e ligas de alumínio, ligas de cobre, chumbo e zinco. Reutilização destes materiais pode reduzir os custos de construção de novos sistemas. Alguns componentes, tais como ouro e cobre são suficientemente valiosos para justificar só por si, a reciclagem destes equipamentos

Os materiais plásticos representam 19 por cento do lixo electrónico total [2]. Um dos problemas com a reciclagem e recuperação de plásticos é a dificuldade em separar as peças por tipo de

plástico. Devido à incompatibilidade de diversos materiais plásticos, as partes desmontadas dos aparelhos devem ser identificadas e classificadas em diferentes tipos de plástico. Estudos demonstram que menos de 1% dos plásticos existente nos aparelhos electrónicos são realmente recuperados para incorporação em novos equipamentos [3].

Reciclagem de impressoras

A Hp, a maior empresa mundial de produtos electrónicos começou em 1998 um programa global de retoma e reciclagem de equipamento, no âmbito do seu programa *Planet Partners*, que foi considerado como o programa mais abrangente de retoma de equipamento no mundo [4]. Neste programa o equipamento é triturado em pedaços pequenos e, posteriormente, separado em seus materiais (aço, alumínio, cobre, plástico, etc). Uma vez separados, estes materiais são enviados para empresas de reciclagem especializadas ou utilizados para recuperação de energia. Greenpeace visitou várias unidades de reciclagem na China e na Índia, e observou que quantidades substanciais de metais pesados tóxicos e compostos orgânicos eram libertados para o local de trabalho e meio ambiente [5].

OBJECTIVOS

Com um interesse renovado na automação e electrónica em ciências agrícolas, o currículo das Licenciaturas e Mestrados da Universidade de Évora incluem agora disciplinas relacionadas com a automação, electrónica e aquisição de dados. A fim de proporcionar aos alunos o equipamento e material necessário para o seu processo de aprendizagem, uma nova abordagem foi utilizada com sucesso: as impressoras são desmanteladas pelos alunos e as peças usadas na sala de aula para desenvolver os seus próprios projectos. Isso tem ajudado a reduzir a pegada ecológica das aulas, e o custo total da realização das aulas de laboratório.

Estima-se que existam pelo menos 1000 impressoras na Universidade de Évora, com uma expectativa de vida de cinco anos. Isso se traduz em um fluxo anual de 200 impressoras descartadas disponíveis gratuitamente para as aulas. Uma das impressoras mais amplamente disponíveis e úteis é a HP série 600 que foi fabricada entre 1994 e 2000.

Este trabalho apresenta os resultados da experiência de quatro anos com desmantelamento e reciclagem das impressoras, descreve os componentes recuperados e alguns usos no ensino de electrónica. Além disso, descreve como este programa pode ajudar o processo normal de reciclagem das impressoras.

RESULTADOS

Cada grupo de dois alunos recebe um conjunto de equipamento, incluindo um microcontrolador, voltímetro, bateria de 12V, painel solar e ferramentas diversas e peças. Para além disso, recebe também uma impressora em fim de vida para desmontar e utilizar as peças. Os alunos conseguem desmantelar completamente a impressora e separar os componentes com base em seu material em menos de 15 min. As 72 peças individuais de uma impressora estão apresentadas na Fig. 1. Na Tabela 1 estão apresentados os componentes constituintes identificados por tipo de material e peso. Note-se que uma vez retirados os componentes reutilizáveis, praticamente todos os demais componentes podem ser separados em materiais simples, prontas para a reciclagem e mistura com material virgem. Cada impressora representa um peso total de cerca de 5000 g, dos quais apenas alguns 500g são materiais indiferenciados que necessitam de separação adicional antes da reciclagem.

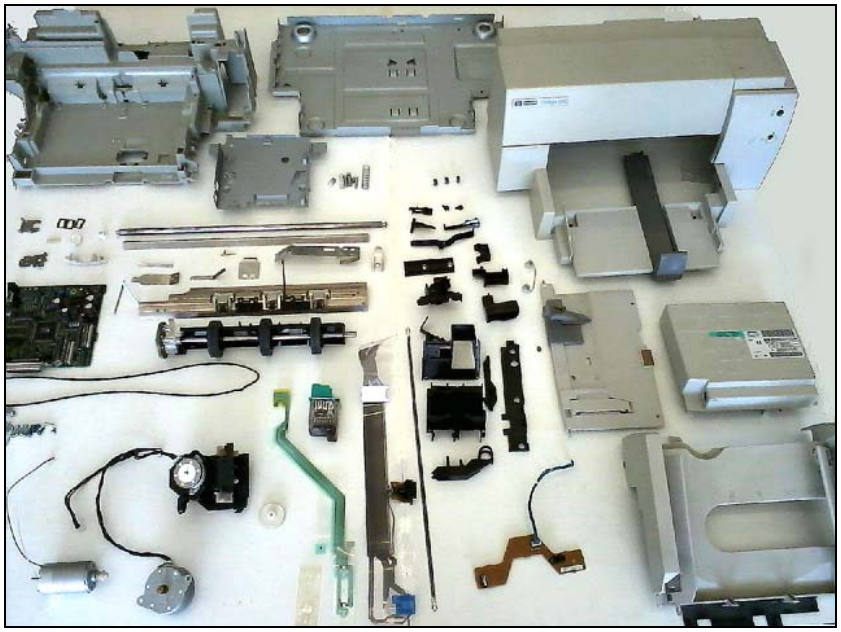


Figure 1. Os componentes de uma impressora HP série 600

Os componentes mais úteis são 2 motores de passos, um motor de 12 V equipado com um carrinho e engrenagens, um termómetro LM35DZ, um relé de contacto, um interruptor com fotodiodo e dois LDRs. Uma consulta às lojas de especialidade revelou que o custo total de aquisição desse equipamento ronda os 70€ (Quadro 2).

Quadro 1. Os componentes de uma impressora HP série 600 discriminados por material e peso.

	Peso, g	***Material***	***Destino***
Componentes reutilizáveis			
Motor PM55L	219,2	Metais ferrosos	Reutilização
Motor PM35L	180,7	Metais ferrosos	Reutilização
Motor C2162-6006	221,1	Metais ferrosos	Reutilização
Placa de sensores	11,4	Comp. electrónicos	Reutilização
Fita + sensor	10,9	Comp. electrónicos	Reutilização
Cabo LPT		Cobre e PE	Reutilização
Componentes ferrosos			
Placa base	1136,2	Liga	Reciclagem
Partes em Aço Inox	415,2	Aço Inox	Reciclagem
Placa traseira	174,6	Liga	Reciclagem
Componentes plásticos			
Plástico preto	134,7	PC 30GF	Reciclagem
Estrutura interna em plástico	411,4	PC 20GF	Reciclagem
Corpo exterior em plástico	1559,6	Policarbonato - PC	Reciclagem
Rolo alimentador	266,9	PC, borracha	Trituração e reciclagem
Electrónica			
PCB	146	Comp. electrónicos	Trituração e reciclagem
Outros			
Peças de plástico e metal	46,5	Mistura plástico e metal	Trituração e reciclagem
Total	4934,9		

A experiência ao longo dos quatro anos demonstra que os alunos tem muito gosto em desmontar o equipamento e retirar os componentes para utilizar nos seus projectos. Por outro lado, eles estão dispostos a fazer o esforço adicional e desmontar e separar os restantes

componentes com base no tipo de material e as marcações nos plásticos, obtendo lotes de material reciclável puro.

Quadro 2. Valor de mercados dos componentes recuperados

Componente	***Preço revenda, € (100 unidades)***	***Fornecedor***
Motor de passos bipolar PM55L	27,40	Digi-Key
Motor de passos bipolar PM35L	22,80	Digi-Key
Termómetro LM35DZ	0,76	Farnell
Motor C2162-6006	12,50	Digi-Key
Interruptor Omron D2F-01FL	1,82	Mouser
Par fotodiodo/ fotosensor	1,75	inmotion
Cabo LPT	2,00	Loja de informática
Total	69,03	

Reutilização dos componentes e subsistemas valiosos

A placa de sensores possui três sensores muito interessantes do ponto de visto de ensino de electrónica: um termómetro integrado de grande precisão tipo LM35, um relé de contacto e um par fotodiodo/fotosensor. O circuito eléctrico da placa está apresentado na Figura 2. A placa pode ser utilizada directamente ou então desmontada para o aproveitamento individual dos componentes.

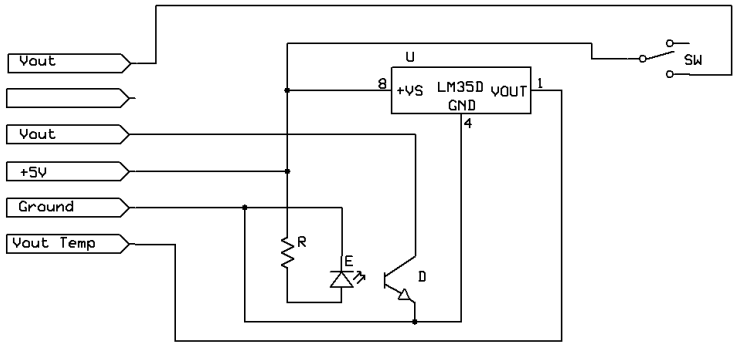


Figura 2. O Circuito eléctrico da placa de sensores C2162 utilizada nas impressoras HP séries 500 e 600.

LM35DZ

Os termómetros integrados do tipo LM35 têm uma precisão de 0,4ºC. O DZ dispõe ainda de uma saída linear de voltagem em função de temperatura, o que é ideal para a sua implementação rápida na sala de aula. O integrado tem um ganho de 0.01V por ºC, o que facilita imenso a leitura de temperaturas com o Conversor A/D incorporado nos microprocessadores AVR.

Um dos primeiros projectos no curso é normalmente a montagem de um termómetro digital utilizando o termómetro LM35. Os outros componentes necessários para realizar este projecto são um LCD e um microcontrolador. A Figura 3 apresenta um termómetro digital construído utilizado apenas uma placa C2162. É possível acrescentar muitas funções utilizando os dois interruptores, tais como a indicação da temperatura média quando se carrega no relé de contacto.

Interruptor fotodiodo/fotoreceptor

Um outro componente interessante na placa de sensores é o interruptor com saída fototransistor, que é utilizado pela impressora para detector o papel. É construído por um emissor LED no infravermelhos próximo (850-940nm), e um sensor. A luz de infra-vermelhos é transmitida num dos lados da abertura e se não haver nada a interromper a sua passagem, é recebida no outro lado da abertura.

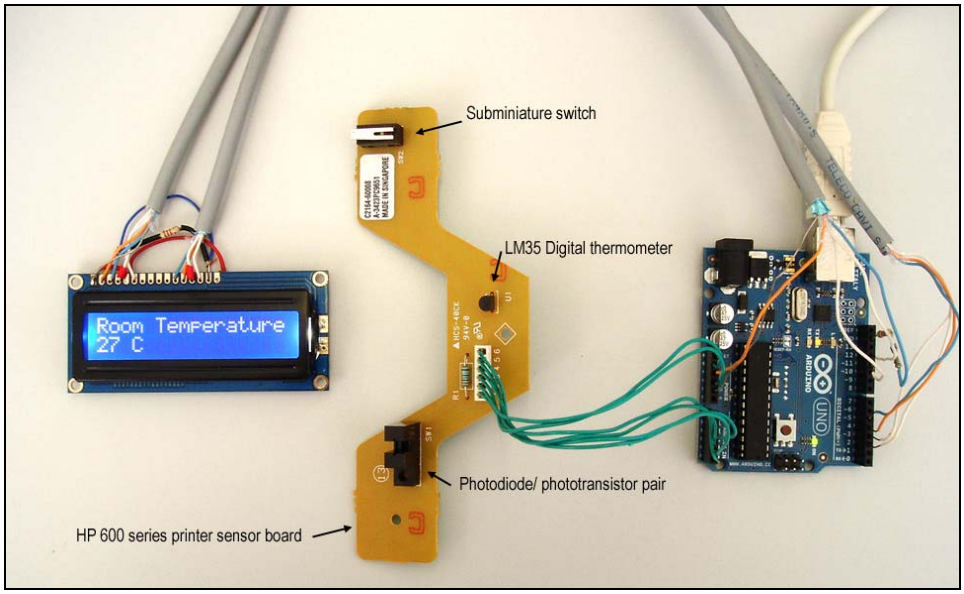


Figura 3. Um termómetro digital construído com uma placa de sensores de uma impressora HP série 600.

Interruptor básico Omron
A Hp instalou um interruptor de baixa força para detector a abertura da tampa da impressora. O interruptor está montado na placa de sensores, mas pode ser retirado facilmente. Nas aulas é utilizado como um interruptor comum, e para o controlo de fim-de-curso na automação.

Motores de passos
Os componentes mais preciosos fornecidos pela impressora são dois motores de passos unipolares: Os motores de imã permanente PM55L e o PM35L fabricados pela NMB. Estes motores de passos podem ser usados em robótica e na automação, uma vez que podem posicionar a cabeça com grande precisão. Nas impressoras HP série 600 o PM35L é usado para limpar a cabeça de impressão, e vem montado com um elevador que proporciona um movimento vertical. Isto é muito útil na visualização e implementação de projectos com motores de passo e permite potenciar a aprendizagem dos alunos. Um integrado com pares Darlington, como por exemplo o UNL2003A pode ser usado para controlar facilmente o motor de passos com o sinal TTL do microcontrolador. O PM55L é o motor de alimentação de papel para a impressora, e é um poderoso motor de passos de1A, que deve ser controlado com um integrado capaz de aguentar a corrente de 1 A, tal como o L298N.

Motor DC
A HP utilize um motor Mabuchi para a deslocação dos tinteiros. Trata-se de um motor de 24V com um torque inicial elevado (1,83 kg-cm) que consegue rapidamente mover os cartuchos durante a impressão. Nas aulas, todo o conjunto é utilizado pelos alunos para construir portões de correr automáticos. Trata-se de um exercício excelente, visto implicar a implementação de rotinas condicionais e de temporização no programa.

Cabo LPT
O cabo LPT é útil no desenvolvimento de aplicações de controlo centrados no PC. A porta LPT1 fornece uma corrente de 5V em cada um dos oito fios do cabo, e este sinal pode ser utilizado para comandar relés de 5V tais como o Finder 40.52, ou para ligar LEDs. Um programa simples realizado numa linguagem adequada, tal como o Qbasic, pode ligar e desligar o relé através do comando "OUT &H378, 255", permitindo o controlo independente de oito aparelhos.

CONCLUSÔES

Os quatro anos de experiência demonstram que os alunos estão dispostos a despender o tempo e desmontar os equipamentos electrónicos para a recuperação de componentes como parte de seu processo de aprendizagem. A partir de cada impressora, os alunos conseguem recuperar motores, sensores e cabos num valor de Mercado de cerca de 70 €. Estes componentes fornecem recursos para a realização de diversos projectos de automação e robótica na sala de aula.

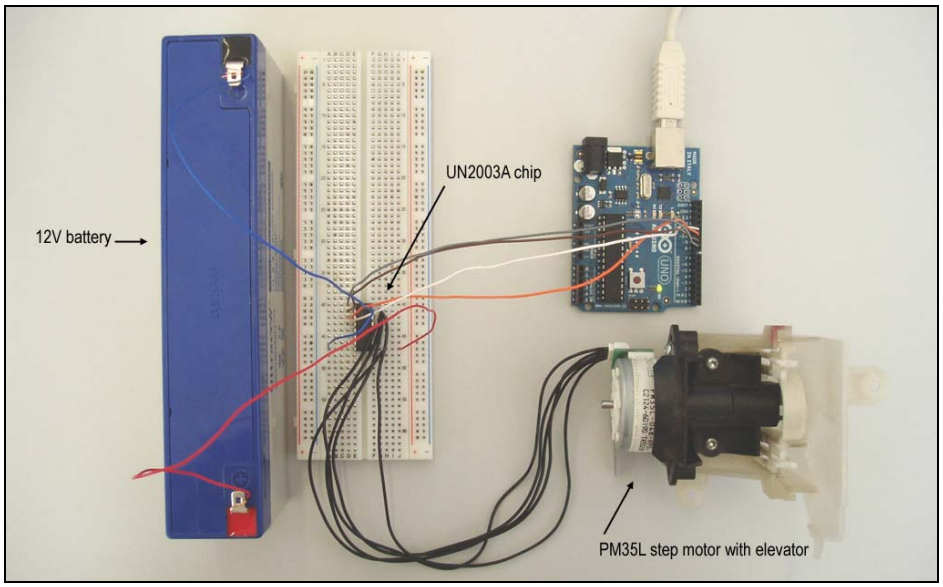


Figura 4. Utilização do motor de passos PM35L e o elevador de tinteiros. Neste caso particular, utilizou-se um integrado UNL2003A para fazer a interface com o microcontrolador.

Os alunos também demonstraram interesse e disponibilidade para fazer um esforço adicional e desmontar e separar os restantes componentes com base no tipo de material e as marcações nos plásticos. Isso resultou na obtenção de 4500 g de material "puro" por impressora, além dos componentes úteis para os seus projectos. A experiência também serviu para despertar ainda mais a preocupação ambiental dos alunos, com muitos deles trazendo o seu próprio equipamento em fim de vida para o desmantelamento na universidade. Outros pediram conselhos sobre como usar o seu equipamento em fim de vida para implementar sistemas de automação simples em suas casas.

Do lado crítico, e analisando os componentes e o design destas impressoras verifica-se que existe uma grande necessidade dos fabricantes desenvolver e projectar produtos limpos com maior duração de vida, que sejam fáceis de reparar, actualizar e reciclar. Neste caso particular, a forma como os materiais são misturadas, os tipos do parafuso e a dificuldade de aceder ao interior da impressora indicam claramente um esforço deliberado do fabricante para desencorajar a reparação e manutenção do equipamento. Isso parece ser contraditório com os vários programas de protecção ambiental da Hp para incentivar a reciclagem no fim de vida do seu equipamento. Além disso, a actual política de maioria dos fabricantes de impressoras para vender as suas impressoras quase ao mesmo preço que os cartuchos de tinta, incentiva a renovação frequente e desnecessária dos equipamentos e um maior impacto sobre o ambiente e os recursos naturais.

REFERENCIAS

[1] J. Bienvenu. U.S. and Worldwide production color printer 2011-2015 forcast and 2010 vender shares. IDC Doc#229205. 2011.

[2] S. Klatt, Recycling personal computers, in. Computers and the environment: Understanding and Managing their Impacts. Edited by Kuehr and Williams, pp 211-229. Kluwer Academic Publishers. 2003.

[3] P.S. Dillon, E.N. Aqua, Recycling Market Development for Engineering Thermoplastics from Used Electronic Equipment; Technical Report 20; Chelsea Center for Recycling and Economic Development, University of Massachusetts: Lowell, MA, 2000.

[4] A. Degher , HP´s worldwide take back and recycling programs: lessons on improving program implementation. International Symposium on Electronics and the Environment. p.224-227. 10.1109/ISEE.2002.1003270 . 2002.

[5] K. Brigden, I. Labunska, D. Santillo, M. Allsopp, Recycling of electronic wastes in China and India: workplace and environmental contamination. Greenpeace Research Laboratories. Technical Note: 09/2005. 2005.

MIPS32 Architecture Simulator

David João Maia and Miguel José Barão

University of Évora, Portugal
m5847@alunos.uevora.pt
mjsb@di.uevora.pt
November 13, 2011

Abstract. The virtualization system is increasingly used in the computer world. Her job entails numerous advantages, and in some cases, achieves better performance than a native machine.
This article seeks to demonstrate the operation of a simulator for MIPS32 architecture, including the "disassembly" of the object code and its representation on the host machine and subsequent execution.
To do this, we will discuss the structure of the processor and the virtual memory, as well as the fundamental mechanisms of functioning, especially the exceptions and interrupts systems.

1 Introduction

In 1981, John Hennessy at Stanford University began work on what would become the first MIPS processor. The initial concept, aimed to dramatically increase processor performance using instruction pipeline, a technique known at the time but difficult to implement.
At the time, the existing architecture followed the model of CISC architecture, consisting of a complex set of instructions that would take several processor cycles to be complete, forcing the processor to spend too much time being on hold in order to execute the next instruction.
The need to increase processor performance RISC architecture was born using a simple and uniform set of instructions. These lost wealth in the amount of possible operations to perform in each statement, but on the other hand, now have a smaller size. The instructions take approximately one processor clock cycle. With the use of pipeline, processors had a noticeable increase in performance since they reuse the dead time of the processor. Currently there are several types of computer architectures and the best known are: x86, x64 or x86-64, PowerPC, SPARC, MIPS, Alpha, ARM, among many others. These architectures are based on the foregoing RISC and CISC. Consequently, these have evolved and added new extensions however its foundations remain the same. With the changing times the need to virtualize systems with different architectures, as mentioned above, grew for the most varied reasons such as cost reduction, energy consumption, use of disk space, increased performance, ease of use and machine migration.

Currently, virtualization is a technique widely used at the enterprise level, reason why the idea of building a simulator for the MIPS32 architecture was born, but also at the same time will aim to deepen the knowledge in the area.
This idea is not entirely original, since there are already other simulators for the MIPS architecture, SPIM and MARS. Although these simulators are simpler, there is other more complex, like QEMU that are able to install the operating system Debian GNU/Linux.
I will use the MARS simulator as an object of comparison, since this is the midpoint between the other simulants already referred. Here is the presentation of a defined structure to represent the host processor in a machine and its main memory.

There are some minimal components required in order to run a simulation that is specifically dedicated to the MIPS32 architecture, requiring only a program that simulates the behavior of main memory, the processor registers and MIPS instruction set. With the components shown to work, you can perform relatively complex programs, with access to the main memory and recursion, such as factorial or Fibonacci sequence, but all in memory.
Called attention to the fact that components such as hard disk, cache, and others are not mandatory to run a simulator, but of course are desirable.
Despite the fact that it is possible to run complex programs, it is not enough to be able to install and run a minimalist kernel (minimal system) or even allow communicate with other peripherals besides the main memory.
Therefore, I will illustrate the operation of the exceptions mechanism, which aims to solve implementation problems preventing the kernel to "crash" or make system calls. Thus I will illustrate the operation of the interruption mechanism, whose conception has been designed to allow communication between the processor and peripherals, including keyboard.

2 Structure

To implement the simulator I decided to use the C/C++ since it is a powerful language that lets you work easily at low level, as opposed to MARS that was developed in JAVA.
So you can run the simulator in any system, as long as it is possible to compile the source code into binary for the machine host. The following is a brief description of the components and data structures used to represent information in memory on the machine host.

2.1 Processor

The main component to be built is the processor since it is here that all information is processed. This requires creating a structure, a struct in this case, to save a set of information including an array to store generic data records, an array of records for monitoring the implementation of the coprocessor zero, the

special registers HI, LO, PC and finally a pointer to a structure that represents the main memory.
The second phase is the implementation of instructions that must run on the processor. All instructions have the same signature, FUNCTION NAME (CPU * cpu). This is due to the fact that all instructions need to access memory to read the 32bit instruction opcode and do the decoding.
To improve the efficiency of the implementation of the simulator, I decided not to use a complex switch with multiple comparisons, but pointers to functions. As the MIPS groups instructions within blocks of instructions, also called opcode encoding, the decoding is much simpler.
When the decoding process is started, the first 6-bits MSB are read from the instruction opcode, which identifies whether it is a direct or indirect instruction. For example ADDI is a instruction part of a subset of instructions, unlike the ADD instruction. Pointers to functions have the following signature:

```
void (*special2[64])(CPU_32 *cpu) = { inst0, ... , inst63 }
```

The structure of the CPU which points to the memory is not fixed, its implementation may change as we shall see later. Macros are used to define the type of memory, so you can add a layer of abstraction to the implementation of memory. It is possible to have multiple implementations of memory, which is utilized in the same way.

2.2 Main Memory

As stated above, the memory is a necessary element for the operation of the simulator. In the starting of the simulator the whole kernel is loaded into memory, respectively, instructions, handlers, drivers and libraries.
The main objective is to get a Translation Lookaside Buffer to run, but its implementation is quite complex. Because of this, I decided to use an implementation in blocks to represent the memory that consists of a list of blocks with base addresses.
The vision of the memory access is always the same, with the signature:

```
WORD load_word(RAM List ram, WORD address, int offset)
void write_word(RAM List ram, WORD address, int offset, WORD info)
```

to read and write data in memory, 32bit, respectively. So through macros, we can decide what type of implementation is used, allowing to develop new ways of drawing a memory implementation.
Despite the fact, that the STACK is totally dependent on the main memory. STACK officially begins at 0x7FFF.FFFF address, but since this is not multiple of base 2, the stack pointer gets booted with the value 0x7FFF.EFFC address, recalling that the STACK grows downward and the HEAP grows upward in the memory addresses.

2.3 Endianness

An important aspect of the architectures is the way it is done the ordering of bytes, this technique is called the endianness, which can be either Big Endian or Little Endian.
If architecture is big endian, then the byte 0 is the MSB (Most Significant Byte), e.g. the leftmost byte. If Little Endian, then byte 0 is the LSB (Least Significant Byte), respectively the rightmost byte.
The endianness as well as natively, influences how the simulator will interpret and organize the object code of the kernel. Once the information is represented in the internal memory of the host machine, this problem disappears because all information will be compiled and executed on the host machine.

3 Operation of the Simulator

3.1 Main execution

Once the simulation starts, you must initialize the CPU and memory structures, followed by their feeding data. All the kernel is loaded into memory for pre-defined addresses, in particular, the instructions to the address 0x0040.0000, heap variables to 0x1004.0000, variables on the stack to 0x7FFF.EFFC since 0x7FFF FFFF is not multiple of base 2.

After the information is loaded into memory, the simulator is ready to begin executing code, the instructions are all performed in an endless while loop.
Thus we do not have to worry about the amount of instructions to be executed, despite the fact that some blocks of instructions loaded into memory may never be executed, in particular exception code handlers or system libraries. The execution flow is controlled by two registers in the processor, the PC (Program Counter) and Inst (current instruction), the first stores the address of the next instruction in memory and the second stores the address of the current instruction. In the main loop is necessary to create a state machine, thus separating two modes of operation, NORMAL and BRANCH DELAY modes.

When conditional or unconditional instructions are executed, like jumps, for example JAL, J, BEQ among others, the processor changes its mode of operation to BRANCH DELAY.
The reason by which this happens is due to the simulation technique of the pipeline architecture. In pipelining, instructions are divided into five phases, respectively the Instruction Fetch, Read Registers, Arithmetic Logic Unit, Memory and Write Back. When a jump instruction is in phase Read Registers, the following instruction is already in the Fetch Instruction phase, so the PC will change to the address of the jump.
Because the instruction had been loaded into the processor, it will be executed not removed from the processor. It is quite common for compilers to introduce NOP instruction after a jump instruction to solve this problem.

In these cases, the instruction following the jump is always executed in the delay slot, all other instructions run in NORMAL mode. After the execution of the instruction in the BRANCH DELAY mode, the processor returns back to NORMAL mode. This is the only "mark" where we can see the presence of the MIPS32 architecture pipelining in the simulation process.

3.2 Exception System

The normal flow of execution can be interrupted when an exception occurs, which is an event triggered in the system. These events can occur due to runtime errors, system calls or external peripherals such as keyboard. When an exception occurs the system stops running, goes into kernel mode and calculates the address for the handler that contains code to be executed to respond to the event.
MIPS has established a set of addresses, called the entry point, which has specific functions.
These addresses are mapped to tables by the MIPS and should not be changed, as well as offset addresses. The architecture is concerned only to define spaces and entry points to the handlers, so that the programmer can insert the code of the handlers there. Space is limited in memory, so the handler should be brief.

It is the function of the programmer to save the general registers, as well as some values of the control registers of the coprocessor 0, including the EPC, should there be an exception or nested exception. In the simulator when an exception happens, it will stop its normal operation and calls a function "RaiseException", whose aim is to calculate the new address of the PC, starting to run from the handler address.
You cannot have the exceptions system to operate without the coprocessor registers implemented, especially, STATUS, CAUSE, EPC, EBASE and BadVaddr. MARS by default does not use any type of handler, but contains the option to program specific handlers for errors.

3.3 Interrupt System

This technique is used to allow multiple external devices to communicate with the processor and its system. When a device sends information it generates an interrupt event, which in turn is treated as if it were an exception, but with a specific exception code, in this case 0x00.
There are three modes of operation for the interrupt system, including Interrupt Compatibility Mode, Vectored Interrupt Mode and External Interrupt Controller Mode. Like MARS, this technique will be implemented using the MMIO (Memory Mapped Input / Output) with EIC (External Interrupt Controller).

So we can send data through the keyboard into the simulator, each time a key is pressed, the simulator will trigger an interrupt event and read the value sent on record in memory of the simulator as if it was a real peripheral. The same applies to the reverse situation, sending information out of the simulator.

If there are multiple devices to communicate you can add priorities to interrupts that are generated through the Interrupt Priority List. This system requires the implementation of the exception system, so it is a mandatory requirement for its operation. Despite the fact that MARS does not simulate all the architecture elements, there are some steps that are simulated.

3.4 SimulUE vs MARS

Clearly, the state of development of the simulator UE is in an embryonic state, but in "fast" growth. Although MARS is more robust and the oldest, this does not simulate the complete architecture, only a few elements. The following is a list of features available in each of the simulator:

Features	**Simul UE**	**MARS**
Input MIPS32 object code	Yes	No
Simulation with pipeline	Yes	Yes
Simulation without pipeline	No	Yes
Little Endian byte ordering	Yes	Yes
Big Endian byte ordering	No	No
Coprocessor 0 registers	All	Incomplete
General purpose registers	All	All
Execption mechanism	Yes	Yes
Programable handlers	Yes	Yes
Interrupt mechanism	Incomplete	No
MMIO (Memory Mapped Input/Output)	Incomplete	Yes
System Calls	Incomplete	Simulated
Cache	No	Yes
STACK	Yes	Yes
TLB?	No	No

As we can see in the table above, there is still much work to do in order to have a fully working simulator. Not everything is bad, many of the features used in MARS are already implemented and running in the simulator UE.

4 Conclusions

With the following work, it was possible to present an approach for developing a simulator for the MIPS32 architecture, using the programming language C/C++. All the key aspects of the architecture simulation were achived, namely the structure of the processor, the main memory and the exceptions system for event handling.
We define an implementation structure for the execution, which is subdivided into two modes of operation of the processor, allowing the simulation of pipelines. Concerning the interrupts mechanism, it was only defined the way it works, since

its implementation is not complete.
It was also objective compare the structure and features of this simulator with other more advanced, mainly MARS, with advantages and disadvantages of both.

5 Future Work

It would be very interesting to raise the level of implementation of the simulator, allowing its implementation not only to a "mini-kernel", but rather develop an elaborated kernel, like the Debian GNU/Linux for the MIPS architecture.
This simulator, which is intended to run within an operating system, allowing it to simulate the behavior of other operating systems, designed for the MIPS architecture and their applications like the most advanced simulators, VMWare, VirtualBox or QEMU.

References

1. Robert Britton, *Mips assembly language programming*, Prentice Hall, Illustrated Edition, 2003.
2. Farquhar Bunce, Philip and Erin, *The mips programmer's handbook*, Morgan Kaufmann, 1994.
3. Microchip, *Pic32 mx family 5xx/6xx/7xx data sheet*, Microchip Technology Inc, 2009.
4. Inc MIPS Technologies, *Mips32 instruction set quick reference*, 1 ed., 2008.
5. _____, *Mips architecture for programmers, volume i-a: Introduction to the mips32 architecture*, 3 ed., MIPS Technologies, Inc, 2010.
6. _____, *Mips architecture for programmers, volume ii-a: The mips32 instruction set*, 3 ed., MIPS Technologies, Inc, 2010.
7. _____, *Mips architecture for programmers volume iii: The mips32 and micromips32 privileged resource architecture*, 3 ed., MIPS Technologies, Inc, 2010.
8. David A. Patterson and John L Hennessy, *Computer organization and design - the hardware/ software interface*, Morgan Kaufmann, 2005.
9. Dominic Sweetman, *See mips run - second edition*, Denise E. M. Penrose, 2006.

Editor de ecrãs de informação

Mário Gusmão, Ricardo Raminhos
Viatecla SA
mario.gusmao@gmail.com, rraminhos@viatecla.com

and Teresa Gonçalves
tcg@uevora.pt

Universidade de Évora

Resumo A existência de mecanismos de comunicação ágeis com o público de massas, adequados e configuráveis de acordo com necessidades específicas de negócio / sector de actividade, são fundamentais no processo de promoção de uma entidade, serviço ou produto. De forma a minimizar o tempo de resposta a estímulos / necessidades de negócio é fundamental que o processo produtivo requeira a intervenção do mínimo de utilizadores / valências, e se possível, limitado ao próprio gestor do negócio. As soluções de *digital signage* apresentam-se como mecanismos de comunicação apelativos e visuais para os quais é necessária a criação de uma *framework* que possibilite a sua gestão e dinamização de forma simples, pela composição de ecrãs de informação, compostos por diferentes tipos de controlos multimédia. O presente artigo apresenta uma solução, tanto na sua vertente de composição como de interpretação e apresentação dos conteúdos, para a composição de ecrãs de informação aplicados a ambientes *digital signage*.

1 Introdução

Os gestores de ecrãs de informação necessitam de ferramentas que os ajudem a criar e a disponibilizar conteúdos de uma forma fácil sem terem de recorrer a programadores informáticos. Estas ferramentas têm de ser intuitivas e de fácil utilização tornando a disponibilização de conteúdos multimédia tão simples como a escrita de um documento de texto.

Com esta ideia em mente, surgiu a necessidade do desenvolvimento de uma ferramenta que permitisse aos gestores de informação sem conhecimento informático de programação criar ecrãs de informação.

Os ecrãs de informação permitem apresentar visualmente informação multimédia com elementos de vídeo, imagem, texto, gráficos e animações. Os ecrãs de informação podem ser dirigidos a um grupo pequeno ou a um grupo grande de espectadores, podem ser apresentados em ecrãs pequenos ou em ecrãs gigantes. Um ecrã de informação pode conter mais do que uma fonte de informação, e juntar diversos tipos de informação de forma a enriquecer a experiência do utilizador.

Os ecrãs de informação podem conter animações, conteúdos multimédia e *layouts* apelativos de forma a optimizarem a transmissão de informação, ou a melhor captarem a atenção do espectador. A divisão de um ecrã em várias secções permite a transmissão de várias informações ao mesmo tempo, a existência de mais pontos de informação para melhor captar a atenção do espectador e a criação de um ecrã mais rico.

A ferramenta desenvolvida permite:

- Que o utilizador consiga criar e editar ecrãs de informação e ao mesmo tempo observar o resultado do ecrã de informação que está a criar.
- Que o utilizador execute os ecrãs de informação de forma a permitir uma pré-visualização do resultado para identificar possíveis erros ou futuras melhorias.
- Que o utilizador guarde o seu trabalho para o poder retomar mais tarde ou ter uma cópia de segurança.
- Ser acedida através de diversas localizações, através de dispositivos diferentes (Televisões, PCs).

Os ecrãs criados pela ferramenta desenvolvida podem ser integrados na plataforma FutureBoxTV (solução de *digital signage* e Internet/WEB TV da Viatecla), que neste momento não oferece forma de criar ecrãs sem o recurso a um informático.

2 FutureBox.TV

A FutureBox.TV[3] é uma plataforma especializada para a distribuição de conteúdos de vídeo e animação de uma forma fluida, transmitindo uma maior dinâmica quando comparado com conteúdos de natureza mais estática como texto ou imagem. Permite ainda a obtenção de dados a partir de fontes externas tais como RSS e Web Services.

A plataforma fornece aos clientes um sistema de backoffice com controlo de acessos e utilizadores. A plataforma pode ser adaptada às especificações únicas de cada cliente e ainda engloba um conjunto de serviços que possibilitam os processos de gestão de vídeo e se necessário os processos de codificação e mudança de formato.

A plataforma pode ser utilizada como:

- *Digital signage*: associada a pontos geográficos específicos, permite a transmissão de múltiplas emissões (potencialmente diferentes) e com controlo total do conteúdo apresentado. Tal permite a integração de experiências mais ricas que valorizam os espaços físicos em que se encontram integrados.
- Internet/WEB TV: pode ser acedida de qualquer lugar, utiliza um browser web para executar a aplicação, tem um ambiente interactivo onde o utilizador escolhe o que que quer ver e permite múltiplas categorias de informação.

3 Estado da arte

Esta secção engloba alguns editores relevantes à natureza do trabalho sobre os quais foi realizado um estudo para compreender as várias funcionalidades e boas práticas de apresentação a transportar para o editor desenvolvido.

Interface do Photoshop O Photoshop[4] é um editor para trabalhar imagens e criar *layouts*. A interacção entre o utilizador e o Photoshop[6] é realizada através de cliques e arrastamento[1]

[1] Do Inglês, *dragging*

dentro do programa. Todas as funcionalidades do Photoshop podem ser encontradas numa única janela, que pode ser redimensionada e movida.

O Photoshop permite que os utilizadores possam customizar a aplicação para melhor satisfazer as suas necessidades e aumentar o seu desempenho. Permite ainda guardar a forma como a sua área de trabalho se encontra para mais tarde a poder retomar de uma forma fácil.

Interface do Visual Studio 2010 O Visual Studio[9] permite que o utilizador tenha vários documentos abertos e que posicione as várias janelas em posições customizadas. As barras de ferramentas e os vários painéis que se encontram ao dispor do utilizador também podem ser customizados, tanto na posição como no tamanho.

As propriedades dos elementos do IDE são definidas no início de acordo com as definições que o utilizador escolhe e existe a possibilidade de fazer um reset às definições para voltar às definições padrão, caso o utilizador tenha feito alguma alteração de que se arrependa.

Interface do Expression Blend 4 O Blend[7] é um editor que permite ao utilizador adicionar objectos a uma área de trabalho e alterar as suas propriedades para customizar os objectos. A interface do Expression Blend 4[5] é parecida com a interface dos outros produtos existentes no Expression Studio[10] . O Expression Blend fornece as ferramentas necessárias para desenvolver e animar o aspecto de interfaces gráficas em Silverlight ou em WPF.

É importante que o utilizador possa ver o seu trabalho no modo de desenho ou no modo de código. O Expresssion Blend permite que o utilizador tenha a sua área de trabalho dividida com estas duas vistas ou possa optar por uma delas.

O Expression Blend também permite que o utilizador possa alterar a forma como os seus painéis estão arranjados. Contém duas vistas padrão: a vista de Desenho e a vista de Animação. Também é possível esconder os painéis para que a área de trabalho fique maior.

4 Arquitectura geral

Na figura 1 pode-se observar um esquema da arquitectura geral do sistema onde o editor se encontra integrado. Esta permite identificar as diversas aplicações implementados e os componentes externos, com os quais é necessário interagir. Ainda expõe os sentidos dos fluxos de dados e o tipo de dados de cada fluxo.

É necessário que o Editor receba recursos e *templates* do sistema de ficheiros local do utilizador e possa lá guardar *templates*. O Editor também deve permitir aceder e armazenar *templates* existentes na parte de administração da FutureBox.TV[3].

O Editor tem como principal função criar, compor e interpretar *templates*. Quando o editor abre um *template* previamente gravado, interpreta-o e apresenta graficamente a informação nele contida.

Os *templates* criados pelo Editor são executados no Visualizador. O Visualizador está presente em 3 aplicações o Editor, o cliente de *digital signage* e o cliente WEB. No editor o Visualizador tem como principal objectivo fornecer ao utilizador uma forma de prever o trabalho efectuado.

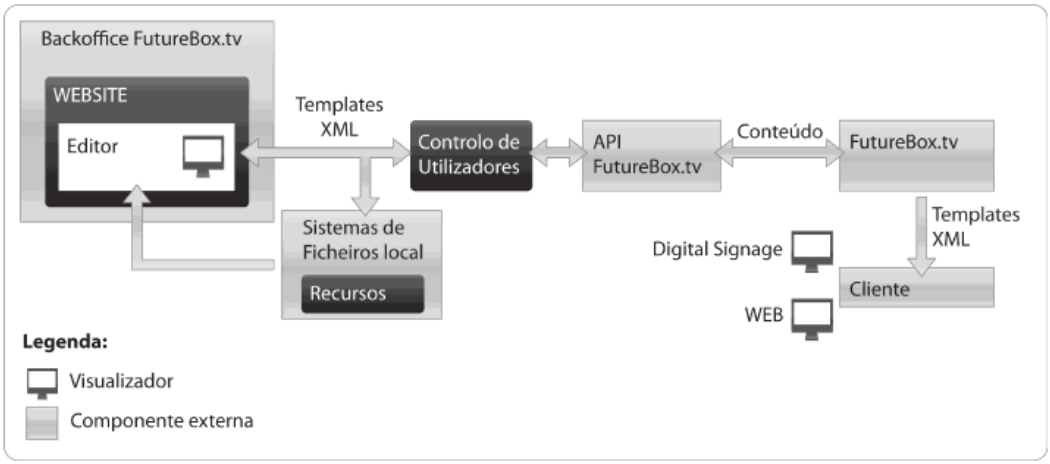


Figura 1. Esquema da arquitectura geral do sistema onde o editor se encontra integrado

No cliente de *digital signage* e no cliente WEB o Visualizador tem como objectivo mostrar os conteúdos.

É necessário um sistema de controlo de utilizadores para validar quem utiliza o Editor no acesso a *templates* existentes na área de administração da FutureBox.TV de modo a garantir que cada utilizador só acede aos seus *templates* e recursos.

4.1 Linguagem declarativa de suporte a aplicação

Foi decidido usar um documento XML[1] pois este permite criar uma estrutura de dados evolutiva e facilmente adaptável que contém toda a informação necessária para guardar o trabalho do utilizador. Quando em execução, o XML é carregado e gravado.

Para validar o documento XML foi decidido usar XML Schema[2] , pois este permite validar os tipos de dados e a estrutura/coerência da informação. Quando se pretende criar um novo tipo de objecto, para que a linguagem o reconheça, basta adicioná-lo à lista de objectos com as respectivas propriedades.

O documento XML é composto por dois tipos de informação principal. O primeiro refere-se a informação do *template* que contém meta-informação, como o nome dos autores, a versão e o nome do *template*. O segundo contém a informação referente aos objectos e às suas propriedades. O Editor necessita de toda a informação, enquanto que o Visualizador só necessita da informação referente aos objectos.

O documento XML contém vários elementos simples com a informação do *template* e um elemento complexo que contém a lista ordenada dos objectos. Cada objecto tem as suas propriedades e estas diferem de objecto para objecto. A ordem dos objectos utilizada para representar a ordem pela qual os objectos são inseridos, de forma a controlar as sobreposições dos mesmos. O último objecto é posicionado à frente e os restantes ficam mais atrás.

A lista de objectos contém todos os objectos do *template*. O primeiro objecto é sempre um objecto do tipo background e só pode existir um objecto deste tipo. A seguir ao background, pode existir um número indefinido de objectos, cada um a representar um plugin no Editor.

4.2 Objectos

Para que os utilizadores consigam criar conteúdos ricos e sem a necessidade de recorrerem a um programador, o Editor tem um conjunto de objectos padrão. Cada objecto tem associado um conjunto de propriedades que permitem ao utilizador personalizar o seu comportamento. A combinação de vários objectos permite aos utilizadores criarem um conteúdo rico.

Foram analisados alguns exemplos de conteúdos para identificar diversos tipos de objectos. A base dos objectos criados, para efeitos de composição, teve como recursos principais texto, vídeo, áudio e imagem. Em alguns conteúdos detectaram-se indicadores de tempo que, para estarem actualizados, precisam de ser executados no momento. De acordo com esta análise, foram criados os seguintes tipos de objectos:

- Texto: utilizado para representar e animar texto.
- Vídeo: utilizado para permitir a reprodução de conteúdos de vídeo.
- Áudio: utilizado para permitir a reprodução de conteúdos de áudio.
- Imagem: utilizado para colocar imagens que permitam enriquecer e tornar o trabalho mais apelativo. Também pode ser utilizado como slideshow para representar um conjunto de imagens
- Data e Hora: utilizado para representar a data e a hora.
- Background: utilizado como base para os outros objectos e para alterar a resolução da área de trabalho.

4.3 Editor

O Editor é a aplicação principal, tendo como função permitir a criação de *templates* de ecrãs de informação. O utilizador pode criar os *templates* adicionando os objectos padrão à área de trabalho. Os objectos podem ser personalizados a partir de uma barra de propriedades que contém as propriedades do objecto seleccionado. O Editor foi desenvolvido em Silverlight[8] e está contido num Web Site existente no backoffice da FutureBox.TV.

O Editor é constituído por uma área de trabalho, uma barra de propriedades, uma barra de menus, uma barra de objectos e uma lista de objectos do projecto. Estes componentes são descritos seguidamente. Na figura 2 podemos ver o Editor e todos os seus componentes.

Área de trabalho. A área de trabalho é o elemento do Editor onde o utilizador foca mais a sua atenção e permite a interacção directa com os objectos, tanto para alterar a sua posição como a sua dimensão. É importante destacar o objecto seleccionado dos outros objectos na área de trabalho e evidenciar as áreas onde o utilizador pode efectuar acções directas sobre ele.

Figura 2. Editor desenvolvido

Barra de propriedades. A barra de propriedades permite visualizar e alterar as propriedades dos objectos e está situada à direita do Editor, ocupando a totalidade da sua altura, pois não necessita de dividir o seu espaço com nenhum painel. A barra de propriedades está sempre visível para permitir a interacção fácil com os objectos presentes na área de trabalho.

Sempre que não seja possível mostrar todas as propriedades de um objecto, é apresentada uma barra de scroll. Quando um objecto é seleccionado, a sua lista de propriedades é apresentada na barra de propriedades.

Barra de menus. A barra de menus permite ao utilizador gravar, visualizar e abrir um *template*. Estas acções não foram integradas nos outros painéis, pois não se encontram realmente no âmbito de nenhum deles. Esta barra situa-se no canto superior esquerdo, à semelhança da sua localização na maioria das aplicações, permitindo ao utilizador encontrá-la de forma intuitiva.

Lista de objectos. A lista de objectos contém todos os objectos criados no projecto e permite aos utilizadores seleccionar um objecto de forma rápida e directa, já que na área de trabalho os objectos sobrepõem-se uns aos outros e os que ficam no topo podem ocultar os que ficam abaixo.A lista de objectos permite alterar a ordem dos objectos por "arrastamento". Também contém a opção para remover os objectos.

Barra de plugins. A barra de plugins contém todos os objectos que podem ser adicionados ao projecto e está situada à esquerda do Editor. Cada plugin é identificado por um nome e por uma imagem. Esta imagem é utilizada como forma de identificação dos plugins nos outros componentes do Editor.

4.4 Visualizador

A função do Visualizador é executar um *template* e exibir a sua informação. É importante que o conteúdo exibido pelo Visualizador ocupe a maior área possível sem perder a proporção. Para conseguir isto, é aplicado um zoom ao conteúdo do Visualizador. Como o zoom é aplicado de forma a manter a proporção do conteúdo, este nem sempre ocupa a totalidade da área disponível. As áreas adjacentes ao conteúdo estarão a preto para lhe dar destaque.

Quando o Visualizador recebe um *template*, começa por validá-lo. Se o *template* for válido, o Visualizador constrói o seu conteúdo percorrendo a lista de objectos existente no *template* e criando cada um dos objectos com as propriedades correctas. Após serem todos gerados, os objectos são adicionados ao conteúdo.

5 Conclusão e trabalho futuro

Este artigo apresenta duas aplicação para manipulação de ecrãs de informação, o Editor e o Visualizador. Estas aplicações contêm um conjunto de objectos base com uma série de propriedades. O Editor permite compor *templates* e o Visualizador permite executar os *templates*. Cada *template* contem um conjunto de objectos e representa um ecrã de informação.

Como trabalho futuro, pretende-se realizar testes de usabilidade para o Editor, recolhendo informação para futuras melhorias e para o desenvolvimento de novos controlos como por exemplo um carrossel de vídeos. Introduzir lógica de animação para controlar a entrada e saída dos vários objectos do *template*, este controle pode ser realizado através de inserção de uma barra temporal de eventos

Referências

1. World Wide Web Consortium. Extensible markup language (xml), Consultado em Janeiro 2011. `http://www.w3.org/XML/`.
2. World Wide Web Consortium. Xml schema, Consultado em Janeiro 2011. `http://www.w3.org/XML/Schema`.
3. FutureBox.TV, Consultado em Janeiro 2011. `http://futurebox.tv/`.
4. S. Johnson and F.P. Inc. *Adobe Photoshop Cs5 on Demand*. On Demand Series. Que, 2010.
5. C. Leeds, E. Kosinska, and M. Inc. *Microsoft Expression Blend 4 Step by Step*. Step by Step. Microsoft Press, 2011.
6. D. McClelland. *Adobe Photoshop CS5 One-on-One*. O'Reilly Series. O'Reilly Media, 2010.
7. Microsoft. Expression blend, Consultado em Janeiro 2011. `http://www.microsoft.com/expression/products/blend_overview.aspx`.
8. Microsoft. Microsoft silverlight, Consultado em Janeiro 2011. `http://www.silverlight.net/`.
9. Microsoft. Visual studio, Consultado em Janeiro 2011. `http://msdn.microsoft.com/en-us/vstudio/aa718325`.
10. Microsoft. Expression studio, Consultado em Novembro 2011. `http://www.microsoft.com/expression/`.

Subscrição de Conteúdos de Vídeo e Visualização em Ambientes Ricos

David Caeiro, Ricardo Raminhos
Viatecla SA
dcaeiro@viatecla.com, rraminhos@viatecla.com

and Teresa Gonçalves,
tcg@uevora.pt

Universidade de Évora

Resumo Actualmente existe um grande número de conteúdos vídeo *online* disponíveis ao utilizador. No entanto, muitas vezes esse acesso é restrito se não for numa lógica de videoclube. Noutros casos, o acesso a vídeos gratuitos de prestação de serviços está muitas vezes limitado ao ambiente *Web*. Este trabalho apresenta uma aplicação de subscrição de vídeo gratuito para o ambiente do Windows Media Center, que utiliza a FutureBox.TV como repositório de conteúdos vídeo.

1 Introdução

À medida que aumenta a largura de banda, também cresce o número de conteúdos vídeos disponíveis ao utilizador na Internet. Hoje em dia existem diversos ambientes ricos de pesquisa e interacção com conteúdos vídeo, muitas vezes associados à sua apresentação na perspectiva de cinema em casa[1]. Estas plataformas são suportadas por aplicações proprietárias/específicas de certas *setup boxes* (caso da ZON e MEO), ambientes integrados de apresentação de media, como o Microsoft Media Center, ou associados a dispositivos que, mesmo não tendo como principal objectivo a apresentação de vídeo, possuem esta funcionalidade (consola Xbox 360). No entanto, o acesso a conteúdos gratuitos sem ser numa lógica de vídeoclube é muitas vezes restrito, principalmente quando se tratam de vídeos em alta definição. Numa outra vertente, existem plataformas de distribuição de vídeo que contendo múltiplos conteúdos, geralmente gratuitos para divulgação e apresentação na Internet como serviço público, encontram-se limitadas ao ambiente *Web*, não dando acesso ao utilizador final através de outros formatos.

Existe assim a necessidade de disponibilizar um serviço que permita a subscrição de conteúdos vídeo gratuitos de alta qualidade a partir de um repositório especializado de media e a expansão desse serviço para outros ambientes, que não a *Web*. Este serviço possibilita ao utilizador visionar esses conteúdos em diferentes situações, aumentando assim o "raio de acção" das várias entidades que disponibilizam os seus vídeos nesses repositórios.

[1] Do inglês *Home Cinema*

Este artigo encontra-se estruturado da seguinte forma: a secção 2 descreve alguns conceitos tidos em consideração no desenvolvimento da aplicação, bem como as ferramentas utilizadas; a secção 3 apresenta os requisitos definidos para a aplicação e a arquitectura da mesma; a secção 4 descreve dois produtos relacionados com a aplicação desenvolvida; na secção 5 encontram-se as conclusões alcançadas com o desenvolvimento da aplicação e o trabalho futuro.

2 Conceitos e Ferramentas

Esta secção engloba alguns conceitos relevantes à natureza do trabalho, bem como as ferramentas usadas no seu desenvolvimento.

2.1 Conceitos

Sistemas Multimédia Os media electrónicos são uma parte essencial da multimédia. Em meios naturais, a informação viaja na forma de ondas de luz e ondas sonoras. Os meios electrónicos distinguem-se dos meios naturais pela sua capacidade de gravar, duplicar, manipular e sintetizar a informação.

O transdutor é um dispositivo capaz de mudar o formato dum sinal. Como exemplos destes dispositivos temos os microfones e câmaras de vídeo, que convertem as ondas sonoras e intensidade de luz, respectivamente, em sinais eléctricos, e os altifalantes e monitores que convertem sinais eléctricos em luz e som, respectivamente [6].

A media digital tira partido dos avanços das técnicas de processamento computacionais e herda os pontos fortes dos sinais digitais. As características que os tornam superiores aos meios analógicos são:

- Robustez: a qualidade dos media digitais não se degrada à medida que são feitas cópias. São mais estáveis e mais imunes ao ruído e erros que ocorrem durante o processamento e transmissão. Os sinais analógicos sofrem de atenuação do sinal à medida que se vão fazendo cópias e são influenciados pelas características do próprio meio.
- Integração perfeita: a integração de diferentes media através de armazenamento digital, processamento e tecnologias de transmissão, independentemente das propriedades do media. Portanto, a media digital elimina a dependência do dispositivo num ambiente integrado e permite uma fácil composição de dados e uma edição não-linear.
- Reutilização e permutabilidade: com o desenvolvimento de normas para os formatos de troca mais comuns, a media digital tem maior potencial para ser reutilizada e compartilhada por vários utilizadores.
- Potencial para fácil distribuição: milhares de cópias podem ser distribuídas electronicamente através dum comando simples.

Streaming* vs *Download O *streaming* e o *download* são dois métodos distintos para a obtenção de vídeos, cada um com as suas vantagens e limitações. Existem diferentes tipos de *streaming* e de *download*, mas irão ser focados apenas os mais usuais: o *streaming* tradicional e o *download* tradicional.

O *download* tradicional foi a primeira forma de entrega a ser utilizada e a mais popular. Para utilizar este formato o cliente tem de descarregar a totalidade do vídeo para o disco rígido do computador para que possa posteriormente ser visualizado localmente com o programa de reprodução. Tal como o *streaming*, é necessário ter o conjunto de *codecs*[2] certo instalado. A grande desvantagem deste método é que o utilizador só pode visualizar o vídeo quando este estiver totalmente descarregado e o tempo de *download* depende muito do tamanho do vídeo. No entanto, com o *download*, enquanto o utilizador mantiver o vídeo no computador, este pode ser visualizado a qualquer altura sem ter que ser transferido novamente. Nos *downloads*, os servidores são simples, com suporte a HTTP[3] ou FTP[4].

No *streaming* tradicional existem diferentes protocolos de entrega, mas ir-se-á focar apenas o protocolo RTSP[5] que por utilizar sessões sobressai em relação aos restantes já que atribui ao utilizador o controlo da transferência. Uma sessão é iniciada no momento em que um utilizador se liga ao servidor e dura até que essa ligação seja terminada; durante essa sessão o utilizador tem a capacidade de controlar a reprodução e a transferência através de mensagens que serão enviadas na sessão aberta. Tais mensagens são pré-definidas, como é o caso do *Pause* e *Play*. Nestas sessões, os dados são codificados em pacotes usando o protocolo UDP[6] , o que pode provocar o bloqueio da transmissão por *firewalls* e *proxies*, prejudicando a qualidade de serviço. Sempre que se inicia uma sessão, o servidor envia uma lista chamada *Presentation* que indica ao programa de visualização do utilizador quais os conteúdos disponíveis e ainda informações sobre esses conteúdos, tais como descrições dos vídeos e *codecs* utilizados. Os vídeos disponíveis num servidor podem ter as componentes de som e vídeo separadas em diferentes servidores de modo a dividir a carga.

2.2 Ferramentas

As ferramentas a seguir descritas são as mais importantes para o desenvolvimento da aplicação, cada uma com a sua função: Windows Media Center (ambiente de execução), Windows Media Center SDK (desenvolvimento) e FutureBox.TV (repositório).

Windows Media Center O Windows Media Center [3] é um centro multimédia desenvolvido pela Microsoft que engloba várias funcionalidades mul-

[2] Acrónimo de Codificador/Descodificador, dispositivo de hardware ou software que codifica e/ou descodifica sinais.
[3] *HyperText Transfer Protocol.*
[4] *File Transfer Protocol.*
[5] *Real-time Streaming Protocol.*
[6] *User Datagram Protocol.*

timédia e que permite realizar funções como ver e gravar programas de televisão e filmes, ver DVDs e reproduzir ficheiros de música, CDs de música ou ouvir rádio. Os ficheiros multimédia podem estar situados em discos rígidos, drives ópticas, locais na rede ou em serviços como o Netflix.

De modo a conseguir reproduzir e gravar programas de televisão através de antena normal, satélite ou cabo, o Windows Media Center usa dispositivos de sintonização de televisão (placas de TV) e os programas gravados podem ser posteriormente enviados para dispositivos multimédia móveis ou gravados para DVD. Caso não haja possibilidade de ligar o computador directamente à televisão, é possível fazer *streaming* desses programas gravados ou que estejam a ser transmitidos ao vivo para outros dispositivos que servem como extensão do Windows Media Center[7] (como a Xbox 360), com o objectivo de o visionar numa televisão. O Windows Media Center tem ainda a capacidade de organizar e mostrar todas as músicas, imagens e vídeos encontrados tanto no disco rígido como em unidades de armazenamento amovíveis e noutros computadores da rede.

Um dos objectivos do desenvolvimento do Windows Media Center foi criar uma plataforma programável à qual pudessem ser anexados novas aplicações usando o SDK[8] respectivo.

Microsoft Media Center SDK 6.0 Sendo o Microsoft Media Center uma plataforma expansível, é possível criar novas aplicações e serviços para adicionar capacidades e experiências novas ao utilizador.

O Microsoft Windows Media Center SDK permite criar serviços e aplicações multimédia ricas e é composto por um conjunto de documentação de programação, amostras de programas, *templates* e guias do AUI[9], entre outros recursos necessários para o desenvolvimento de aplicações para Windows Media Center (estas aplicações podem ser controladas por rato, teclado ou comando remoto).

As aplicações Windows Media Center usam as seguintes tecnologias para controlar e expandir as funcionalidades do Media Center: Microsoft .NET Framework (um ambiente de desenvolvimento e execução que permite combinar diferentes linguagens de programação de modo criar aplicações e serviços baseados no Windows); API do Windows Media Center (que permite criar funções automatizadas tal como sintonizar para programas de TV ao vivo e controlo parental); MCML, uma linguagem declarativa baseada no XML usada para definir a interface de utilizador.

FutureBox.TV A FutureBox.TV [5] é uma plataforma da Viatecla especializada para a distribuição de conteúdos vídeo, geralmente de forma gratuita. Esta plataforma foi desenhada para ser extensível através da ligação a bases de dados

[7] Do inglês *Windows Media Center Extenders* ou *Extender for Windows Media Center*.
[8] *Software Development Kit*.
[9] *Application User Interface*.

externas que podem ser interpretadas por *templates*[10] de animação capazes de ler esses dados.

A plataforma FutureBox.TV é gerida através de um *BackOffice Web*. Os modelos de dados, *workflows* e processos de controlo de acesso encontram-se na componente FutureBox Server e são suportados pelo gestor de componentes e *workflow* Viatecla Scriptor Server. O modelo de dados é traduzido para uma base de dados relacional que é mapeada em SQL Server.

A FutureBox.TV tem três vertentes: a vertente de Internet, a vertente *Corporate* e a vertente de repositório especializado de distribuição de conteúdos vídeo. Na vertente de Internet o cliente acede à FutureBox.TV através dum navegador de Internet com recurso à tecnologia Silverlight. Ao aceder à FutureBox.TV é iniciada a apresentação de conteúdos vídeo. Estes conteúdos estão definidos numa lista de reprodução.

Na vertente *Corporate*, a apresentação dos conteúdos está associada a postos geográficos físicos onde não há interacção por parte do utilizador. Os conteúdos a serem visionados são definidos pelo administrador através do *BackOffice*. Um cliente é composto por um PC ligado a um ou mais LCDs de modo a apresentar os conteúdos.

Na componente de plataforma especializada em vídeo, a FutureBox.TV funciona como repositório de metadados e conteúdos que estão disponíveis através dum conjunto de serviços API de modo a possibilitar o acesso por aplicações externas.

3 Arquitectura do Sistema

3.1 Requisitos

Requisitos Funcionais Os requisitos funcionais que se pretendem da aplicação são os seguintes:

- Lista de categorias: Deverá ser apresentado ao utilizador uma lista das categorias de vídeos existentes em catálogo para que este possa mais facilmente localizar os vídeos que lhe interessam.
- Descrição do vídeo: Ao seleccionar um vídeo, deve ser apresentado ao utilizador uma breve descrição do conteúdo do vídeo.
- Informações do vídeo: Mostrar informações acerca do vídeo tal como duração e tamanho.
- *Preview*: Adicionalmente, para além da descrição do vídeo, o utilizador deverá ter acesso a um *preview* em baixa resolução do filme de modo a que melhor decida se quer realizar o *download* desse vídeo.

Requisitos Não-funcionais Os requisitos não-funcionais que se pretendem da aplicação são:

[10] Modelo de página pré-definido.

- Interface *user friendly*, de fácil compreensão.
- Capacidade de gestão e retoma de *downloads*: o *download manager* deverá ser capaz de gerir os *downloads* a realizar e ter tolerância a falhas, sendo capaz de retomar o *download* no ponto em que parou no caso de falha de rede ou do utilizador encerrar a aplicação.
- Downloads locais: os *downloads* dos vários vídeos deverão ser feitos para o disco rígido do PC que estiver a executar Media Center.
- Integração no Media Center SDK: a aplicação terá que ser integrada no Media Center SDK para que possam ser executadas no Windows Media Center.
- Interoperabilidade: a aplicação e a FutureBox.TV terão que comunicar entre si na perfeição de modo a trocar informações e pedidos.

3.2 Desenvolvimento da Aplicação

A arquitectura deste trabalho será explicada juntamente com a arquitectura externa envolvida, uma vez que estas interagem uma com a outra. A arquitectura implementada liga à arquitectura da FutureBox.TV já existente, tal como demonstrado na figura 1, para a aquisição de metadados e ficheiros multimédia.

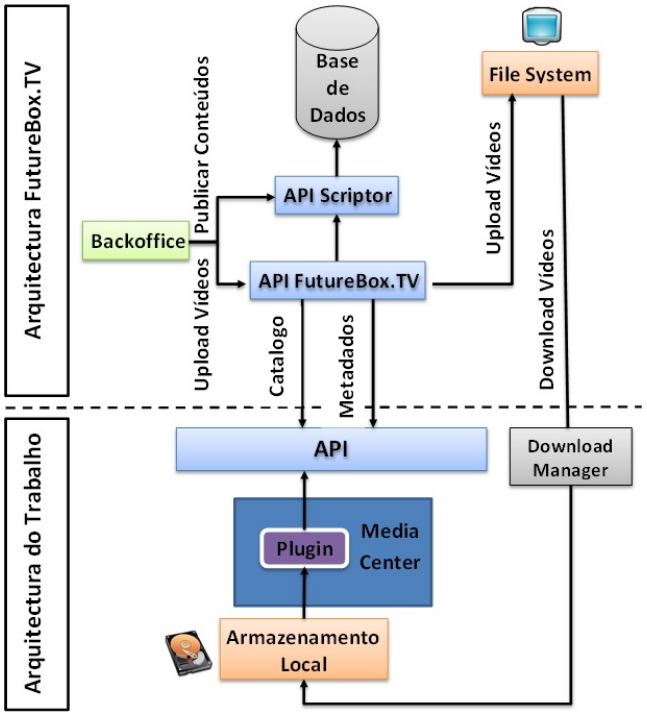


Figura 1. Arquitectura do trabalho.

A aplicação (*plugin*) do Media Center efectua os pedidos à API da FutureBox.TV através de uma API em JSON[11] desenvolvida para o efeito. Os dados a serem apresentadas no *interface* da aplicação (figura 2) são obtidas através da API da FutureBox.TV. A API da aplicação faz o pedido e a API da FutureBox.TV envia de volta um segmento de código JSON com as informações de catálogo (quais os vídeos disponíveis) e os metadados de cada vídeo. Através da API da aplicação são também efectuados os pedidos para o *download* de vídeos. Esses vídeos encontram-se no *File System* da FutureBox.TV e são transferidos directamente desse *File System* para o disco rígido do PC. De modo a garantir que o *download* seja realizado nas melhores condições possíveis é usado um *Download Manager*[12] para garantir que em caso de interrupção do *download*, este possa ser retomado no ponto onde foi interrompido, sem necessitar de voltar a fazer o *download* total. Esta gestão dos downloads é realizada pelo BITS[13], um sistema de gestão de transferência de ficheiros usado pelo Windows para gerir a transferência de actualizações do sistema.

É também no *File System* que se obtêm as pré-visualizações dos vídeos antes de se efectuar o download para os armazenamentos locais.

Figura 2. Exemplo do ecrã principal da aplicação apresentando o catálogo.

[11] *JavaScript Object Notation*: formato leve para a troca de dados computacionais derivado da linguagem JavaScript.
[12] Gestor de *downloads*.
[13] *Background Intelligent Transfer Service*.

Em termos de *upload* e publicação dos conteúdos, tal é realizado através do *backoffice* da FutureBox.TV. Os pedidos de upload de vídeos são feitos à API FutureBox.TV e esta faz o upload para o *File System*. Antes de realizar o upload, a API da FutureBox.TV faz um pedido ao API do Scriptor para que este crie metadados e uma referência ao *File System* na base de dados. Ao contrario do *upload* de conteúdos, o pedido para a publicação de conteúdos é feita do *backoffice* directamente à API do Scriptor.

4 Trabalho Relacionado

Seguidamente são apresentados dois dos maiores serviços de subscrição de conteúdos de vídeo. A aplicação desenvolvida apresenta claras vantagens em relação a estes serviços, nomeadamente:

- Download de vídeos para o disco rígido, o que permite serem visionados a qualquer altura.
- A subscrição dos vídeos é gratuita, não necessitando de pagamentos adicionais para aceder á totalidade dos conteúdos disponíveis.
- Há uma agregação de conteúdos vídeo de domínio publico na mesma plataforma.

4.1 Netflix

O Netflix [1] é um dos maiores serviços de aluguer de filmes existentes. Foi criado em 1997 por Reed Hastings e Marc Randoph e entrou em funcionamento em Abril de 1998. O serviço começou por fazer o aluguer de filmes em DVD pelo correio, sendo uma das primeiras companhias a usar esse sistema. A 16 de Janeiro de 2007 a Netflix anunciou que iria disponibilizar o seu serviço por *streaming*, ao mesmo tempo que iria expandir essa tecnologia para "todos os ecrãs com ligação á internet, desde telemóveis a PCs a ecrãs de plasma" [1].

O Netflix disponibiliza uma API [4] para os utilizadores poderem, caso o assim o desejem, integrar o Netflix nas suas próprias aplicações, sejam elas para *web*, dispositivos móveis ou televisão. Os serviços disponibilizados pela API englobam realizar pesquisas de filmes e séries, obter o catálogo de filmes e detalhes de cada filme, incluir as funcionalidades dos botões Adicionar e Reproduzir e mostrar as pontuações dadas pelos utilizadores a um certo filme ou série.

O Netflix está disponível nas plataformas iOS, Android, Web, PlayStation 3, Xbox 360, Windows Media Center e através da televisão.

4.2 Hulu

O Hulu[7] é um *website* de subscrição de serviços que oferece *streaming* de vídeos de séries, filmes e programas televisivos. Foi criado em 2007 através da associação dos canais NBCUniversal, Fox Entertainment Group e Disney-ABS Television Group, como medida para combater a pirataria online. Comparativamente ao

Netflix, o seu catálogo de filmes não é muito extenso, dedicando-se mais a programas televisivos, onde supera a concorrência.

Apesar de ser um serviço gratuito, para ser acedido noutras plataformas que não o *browser* do PC ou Mac (para além de ter acesso a conteúdos exclusivos), é necessário subscrever o Hulu Plus[2], a versão paga do serviço. O Hulu Plus está disponível nos ambientes *web*, iOS, Android, PlayStation 3 e, recentemente, na Xbox 360.

5 Conclusões e Trabalho Futuro

A aplicação apresentada neste artigo oferece a possibilidade do utilizador consultar um catalogo de vídeos de alta qualidade existentes no repositório, ver uma amostra de baixa qualidade do vídeo, fazer o download gratuito desses vídeos e gerir a sua biblioteca pessoal.

Com este trabalho, conseguiu-se uma aplicação que se destaca das restantes aplicações na área por permitir download gratuito de vídeos para o disco rígido, sem a necessidade de pagamentos adicionais para aceder á totalidade dos conteúdos disponíveis e em que há uma agregação de conteúdos vídeo de domínio publico na mesma plataforma.

O trabalho futuro inclui a realização de testes de usabilidade e performance de modo a melhor avaliar o desempenho das funcionalidades da aplicação.

Referências

1. R. Chiu, S. Doroudi, T. Haussler, and A. Khosla. Netflix: Entering the video on demand industry through providing streaming movies, 2007.
2. J. Kilar. Introducing hulu plus: More wherever. more whenever. than ever, Consultado em Maio de 2011. `http://www.fireproductions.com/pdf/news/49.pdf`.
3. Microsoft. About the windows media center sdk, Consultado em Abril 2011. `http://msdn.microsoft.com/en-us/library/aa468227.aspx`.
4. Netflix. Welcome to the netflix developer network, Consultado em Abril 2011. `http://developer.netflix.com/page`.
5. Viatecla. Futurebox.tv, Consultado em Maio 2011. `http://www.viatecla.com/futurebox`.
6. O. Victor and L. Wanjiun. Distributed multimedia systems. *PROCEEDINGS OF THE IEEE*, 85(7), Julho 1997.
7. Wikipedia. Hulu, Consultado em Maio 2011. `http://en.wikipedia.org/wiki/Hulu`.

NXT MindStorms e Aprendizagem por Reforço

João Coelho
m6416@alunos.uevora.pt

Teresa Gonçalves
tcg@uevora.pt

Universidade de Évora

Resumo A aprendizagem por reforço é uma aprendizagem por tentativa e erro, onde o agente, através da interacção com o ambiente, aprende a realizar uma tarefa com base em recompensas positivas e negativas.
Este artigo pretende analisar o comportamento de um robô implementado com um sistema de aprendizagem por reforço cujo objectivo consiste em seguir uma linha. Para tal, foi utilizado o robô educacional criado pela Lego, o NXT Mindstorms, implementado com o algoritmo Q-learning.
Realizaram-se experiências com o propósito de determinar quais os valores óptimos das variáveis principais do algoritmo Q-learning (taxa de aprendizagem, o factor de desconto e a taxa de exploração), para que o robô tivesse um bom desempenho.
Conclui-se que um robô implementado com um sistema de aprendizagem por reforço consegue aprender uma determinada tarefa em poucas iterações (passos).

1 Introdução

A aprendizagem por reforço é um campo da aprendizagem automática, onde um agente através de tentativa e erro, tenta maximizar a recompensa que recebe ao interagir com o ambiente para realizar uma tarefa específica [3].

Acerca de vinte anos surgiu um enorme interesse na utilização da aprendizagem por reforço na robótica, o que tem beneficiado as investigações relacionadas com estes temas [10].

As tarefas na aprendizagem por reforço são descritas através de funções de recompensa, em vez de instruções específicas para cada situação ou estado. Isto é, ao invés de indicar ao robô qual é a melhor acção para uma determinada situação, o agente através de interacção com o ambiente, aprende qual a melhor acção para essa situação.

Como algoritmo de aprendizagem foi utilizado o Q-learning pois este não precisa de um modelo de ambiente[1].

O artigo está organizado da seguinte forma: a Secção 2 apresenta os elementos da aprendizagem por reforço e o algoritmo Q-learning e a Secção 3 descreve os componentes do robô educacional da Lego NXT Mindstorms, e o ambiente de programação utilizado. A tarefa é introduzida e avaliada na Secção 4. Por último, na Secção 5 são descritas as conclusões.

[1] Do inglês, *model-free*.

2 Aprendizagem por Reforço

A aprendizagem por reforço surge nos anos 60. Em 1961, Minkys publicou um artigo influenciável para esta aprendizagem onde examina alguns dos seus problemas [5].

A sub-secção 2.1 apresenta os conceitos básicos da aprendizagem por reforço e a sub-secção 2.2 descreve os métodos de pesquisa; o algoritmo Q-learning é introduzido na sub-secção 2.3.

2.1 Conceitos básicos

A política, a função de recompensa, a função de valor e o modelo de ambiente são alguns dos conceitos fundamentais da aprendizagem por reforço:

- a política é uma regra estocástica que define o comportamento do agente, ou seja, indica ao agente qual a acção que deve tomar num determinado estado;
- a função de recompensa define o objectivo a atingir num sistema de aprendizagem por reforço. O agente associa um estado (ou par estado-acção) do ambiente a uma recompensa;
- a função de valor indica o ganho total que será acumulado no futuro, quando o agente inicia um determinado estado. Todos os algoritmos de aprendizagem por reforço são baseados na estimativa de funções de valor.
- o modelo do ambiente é algo que imita o comportamento do ambiente, onde existe interacção entre o agente e o ambiente.

Existem dois tipos de funções de valor, a função estado-valor, $V^\pi(s)$, e a função acção-valor, $Q^\pi(s,a)$. Nos processos de decisão de Markov [7], a função $V^\pi(s)$, define-se como

$$V^\pi(s) = E_\pi\{R_t \mid s_t = s\} = E_\pi\left\{\sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \mid s_t = s\right\}, \quad (1)$$

onde $V^\pi(s)$ representa o valor esperado que o agente recebe quando segue a política π, num determinado instante t. A função acção-valor, $Q^\pi(s,a)$, considera o par estado-acção (s,a), e é definida como

$$Q^\pi(s,a) = E_\pi\{R_t \mid s_t = s, a_t = a\} = E_\pi\left\{\sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \mid s_t = s, a_t = a\right\}, \quad (2)$$

onde a representa uma acção no estado s sobre a política π, num determinado instante t.

A função $Q^\pi(s,a)$ calcula as recompensas esperadas com o estado inicial s.

2.2 Métodos de pesquisa

Na aprendizagem por reforço deve existir um equilíbrio entre a exploração[2] e a examinação[3]. Na examinação o agente escolhe a acção com maior recompensa para um determinado estado; à primeira vista parece ser uma boa opção, mas impede o agente de procurar melhores acções.

A exploração permite obter novas informações sobre o ambiente com o objectivo de alcançar melhores níveis de desempenho no futuro, ou seja, o agente explora acções ainda não experimentadas ou estados não visitados, para ter uma visão mais abrangente do ambiente.

Uma das grandes diferenças entre aprendizagem por reforço e aprendizagem supervisionada, consiste no simples facto do agente da aprendizagem por reforço para "aprender"ter que, explicitamente, explorar o ambiente em que se encontra [4].

Existem vários métodos que permitem o equilíbrio entre a examinação e a exploração. Os mais conhecidos são o método ϵ-*Greedy* e o *Softmax*. O método ϵ-*Greedy* na maioria das vezes escolhe a acção com maior recompensa mas, de vez em quando, aleatoriamente é escolhida uma acção independentemente da sua recompensa. A acção com maior recompensa, é escolhida com a probabilidade de $1-\epsilon$, e a acção aleatória é escolhida com a probabilidade ϵ. Isto significa que para a acção a^*, acção com maior recompensa, a política é $\pi_t(s, a^*) = 1 - \epsilon$.

Uma desvantagem do método ϵ-*Greedy* é que ao escolher uma acção aleatória, tanto pode escolher a acção com a melhor recompensa como a acção com a pior. O método *Softmax* foi desenvolvido para ultrapassar esta falha atribuindo um peso para cada acção, de acordo com a sua estimativa acção-valor [2]. Assim, uma acção é seleccionada de acordo com o peso que lhe está associado tornando improvável a escolha da pior acção já que a acção com maior valor de recompensa, tem maior probabilidade de ser escolhida.

Para calcular as probabilidades é utilizada a distribuição de Gibbs, sendo a política dada pela equação

$$\pi_t(a) = \frac{e^{\frac{Q_t(a)}{T}}}{\sum_{b=1}^{n} e^{\frac{Q_t(a)}{T}}} \tag{3}$$

onde T é um valor positivo designado por temperatura. Quando a temperatura for alta, qualquer acção tem (quase) a mesma probabilidade de ser escolhida. No caso de a temperatura ser baixa, essa probabilidade desaparece, fazendo com que as acções sejam escolhidas de acordo com as suas estimativas de valor.

2.3 Q-learning

O algoritmo Q-learning foi apresentado por Watkins em 1989 [9] sendo considerado um dos maiores avanços na área da aprendizagem por reforço já que se

[2] Do inglês, *exploration*.
[3] Do inglês, *exploitation*.

trata de um algoritmo que aprende a política óptima sem criar um modelo do ambiente.

Este algoritmo amplia a função valor tornando-a numa função acção-valor que, para cada estado, armazena o valor de todas acções.

Esta função, $Q^*(s,a)$, expressa, num comportamento óptimo, o valor da acção a realizada no estado s. A relação entre a função valor e a função acção-valor é definida como

$$V^*(s) = \max_a Q^*(s,a) \tag{4}$$

A política é definida de acordo com a equação

$$\pi^*(s) = \arg\max_a Q^*(s,a) \tag{5}$$

e a actualização da função acção-valor é dada pela expressão

$$Q(s_t,a_t) \leftarrow Q(s_t,a_t) + \alpha \left[r_{t+1} + \gamma \max_a Q(s_{t+1},a_t) - Q(s_t,a_t) \right] \tag{6}$$

onde Q é a tabela de valores dos pares estado-acção, a_t é a acção actual, s_t o estado actual, s_{t+1} é o novo estado que resulta da acção no estado actual, $\alpha \in]0,1[$ corresponde à taxa de aprendizagem, $\gamma \in]0,1[$ é o factor de desconto, $\max_a Q(s_{t+1},a_t)$ é a acção com maior recompensa da tabela Q e r_{t+1} é a recompensa que o agente recebe no próximo estado.

Teoricamente a taxa de aprendizagem determina em que medida as novas informações irão substituir as informações antigas. Quando o valor for igual a 0, o agente não consegue aprender nada, contudo se o valor for igual a 1 vai fazer com que o agente só considere apenas as informações mais recentes.

O factor de desconto determina a importância das recompensas futuras. Se o valor do factor de desconto for igual a 0, o agente tira proveito por considerar apenas as recompensas actuais. Contudo, se o factor de desconto for igual a 1 o agente irá lutar por uma recompensa alta, a longo prazo.

3 O Lego NXT Mindstorms

O kit básico do NXT Mindstorms é constituído por um bloco, três motores e quatro sensores (luz, ultra-som, som e toque), mas podem ser adquiridos outros tipos de acessórios.

O bloco NXT, conhecido por *smart brick*, é o "cérebro" do robô. É um bloco controlado por computador que dá vida ao robô, efectuando diferentes operações [1]. Este bloco contém quatro entradas na parte inferior e três na parte superior e uma entrada USB[4].

Os motores permitem ao robô movimentar-se pelo ambiente. Têm a capacidade ajustar a velocidade e calculam com precisão as voltas ou graus que o motor efectuou.

[4] Acrómino inglês para *universal serial bus*.

O sensor de som tem a capacidade de determinar se o nível do som detectado é alto ou baixo e o sensor de toque tem a capacidade perceber se este está a ser pressionado ou não; o sensor de infravermelhos permite ao robô distinguir a luminosidade ambiente e através de um LED[5] vermelho incorporado, consegue detectar a intensidade das cores. Finalmente, o sensor de ultra-som tem a capacidade de medir a distância a que o robô se encontra de um obstáculo.

Para programar o bloco NXT utilizou-se a linguagem lejOS NXJ [6], um projecto open-source que usa *Java Virtual Machine* e fornece uma poderosa *API*[6], assim como as ferramentas necessárias para descarregar o código para o bloco NXT. Esta linguagem foi utilizada por fornecer bibliotecas que suportam várias funções de alto nível, como a navegação e comportamento robótico.

4 Experiências e Avaliação

Esta secção introduz a configuração experimental realizada e apresenta os resultados obtidos.

4.1 Configuração experimental

Para estudar a implementação do algoritmo Q-learning no Lego NXT Mindstorms foi construído um robô capaz de seguir um percurso marcado numa superfície branca.

As sub-secções seguintes descrevem o ambiente, o agente e definem a função de recompensa.

Ambiente. O ambiente é composto por uma superfície onde é desenhado o percurso que o robô deve seguir. A Figura 1 ilustra os percursos utilizados nesta experiência.

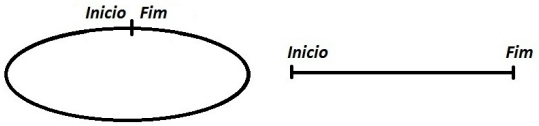


Figura 1: Percursos para o robô seguir

Agente. O robô construído é composto pelo bloco NXT, por dois motores responsáveis pelo seu movimento e por dois sensores de infravermelhos colocados na parte frontal do robô que indicam a sua posição relativamente ao percurso marcado.
A Figura 2 ilustra o robô:

[5] Acrómino inglês para *light-emitting diode*.
[6] Acrómino inglês para *application programming interface*.

Figura 2: Robô capaz de seguir um percurso

O movimento dos dois motores permite realizar quatro acções:
- andar para a frente
- andar para trás
- virar à direita
- virar à esquerda

A velocidade de rotação dos motores é de 6.

Os dois sensores infravermelhos situam o robô num de 4 estados possíveis:
- estado `A`: dois sensores fora da linha
- estado `B`: direito na linha, esquerdo fora da linha
- estado `C`: direito fora da linha, esquerdo na linha
- estado `D`: dois sensores na linha

Função de recompensa. Esta função indica, indirectamente através de valores de recompensa, qual o objectivo que o robô terá de alcançar:

- estado `A`: como o robô tem os dois sensores fora da linha a recompensa deve ser negativa; $r_t = -15$
- estados `B` e `C`: como apenas um dos sensores está fora da linha, a recompensa é maior mas também negativa; $r_t = -5$
- estado `D`: com os dois sensores na linha a recompensa é positiva; $r_t = 10$

Inicialmente foi testado outro conjunto de recompensas, mas o robô não mostrou melhor desempenho. Nessa experiência os valores de recompensa foram $\{r_t(\texttt{A}) = -10, r_t(\texttt{B}) = r_t(\texttt{C}) = 5, r_t(\texttt{D}) = 15\}$.

Algoritmo Q-learning. Como já foi referido, utilizou-se a linguagem *leJOS NXJ* para implementar o algoritmo *Q-learning* com o método de pesquisa ϵ-*Greedy*.

Ao valor inicial da taxa de aprendizagem, α, que pode ser consultada na Tabela 1, é decrementado 0.1 a cada 50 iterações. À taxa de exploração, γ, é também decrementado 0.1 unidades a cada 50 iterações.

Foram realizados testes de combinação dos diferentes parâmetros associados à execução do algoritmo Q-learning, nomeadamente:

– taxa de aprendizagem, α
– factor de desconto, γ
– taxa de exploração, ϵ

Testaram-se todas as combinações dos valores $\{0.3, 0.5, 0.9\}$ em cada parâmetro.

4.2 Resultados obtidos

Para cada combinação de parâmetros foram realizadas 20 experiências com o percurso indicado na secção 4.1.

A Tabela 1 mostra o número médio de iterações (passos) que o robô executa até conseguir fazer um volta completa ao percurso sem que perca a linha e o correspondente desvio padrão.

α	γ	ϵ	**percurso oval**	**percurso linha recta**
0.9	0.9	0.9	300 ± 24	265 ± 16
0.9	0.9	0.5	142 ± 17	119 ± 20
0.9	0.9	0.3	131 ± 31	107 ± 14
0.9	0.5	0.9	330 ± 48	282 ± 22
0.9	0.5	0.5	141 ± 21	117 ± 11
0.9	0.5	0.3	106 ± 20	88 ± 17
0.9	0.3	0.9	215 ± 21	189 ± 21
0.9	0.3	0.5	123 ± 20	102 ± 13
0.9	0.3	0.3	113 ± 21	93 ± 16
0.5	0.9	0.9	300 ± 33	278 ± 23
0.5	0.9	0.5	136 ± 16	113 ± 20
0.5	0.9	0.3	147 ± 37	127 ± 13
0.5	0.5	0.9	389 ± 28	367 ± 11
0.5	0.5	0.5	149 ± 15	123 ± 17
0.5	0.5	0.3	101 ± 18	81 ± 13
0.5	0.3	0.9	482 ± 13	469 ± 19
0.5	0.3	0.5	144 ± 16	129 ± 22
0.5	0.3	0.3	102 ± 11	85 ± 24
0.3	0.9	0.9	589 ± 11	572 ± 16
0.3	0.9	0.5	130 ± 14	113 ± 12
0.3	0.9	0.3	157 ± 20	138 ± 14
0.3	0.5	0.9	357 ± 14	329 ± 17
0.3	0.5	0.5	117 ± 18	95 ± 11
0.3	0.5	0.3	**85** ± 12	**65** ± 24
0.3	0.3	0.9	304 ± 12	286 ± 13
0.3	0.3	0.5	114 ± 18	97 ± 21
0.3	0.3	0.3	111 ± 11	93 ± 14

Tabela 1: Média e desvio padrão do nº de iterações

Pela observação da tabela, é possível constatar que o robô aprendeu mais rapidamente o comportamento desejável na configuração com $\{\alpha = 0.3, \gamma = 0.5, \epsilon = 0.3\}$ para os dois percursos; por outro lado, a configuração com $\{\alpha = 0.3, \gamma = 0.9, \epsilon = 0.9\}$ apresenta os piores resultados para os dois percursos.

Através dos valores apresentados na tabela é possível confirmar que valores altos da taxa de exploração ($\epsilon = 0.9$), para os dois percursos, produzem um comportamento errático do robô com um nº de iterações superior a 300, excepto quando o factor de desconto (γ) é baixo cujo valor médio é 215, para o percurso oval, e 189 para o segundo percurso, linha recta, (configuração $\alpha = 0.9$, $\gamma = 0.3$ e $\epsilon = 0.9$). Este resultado era expectável já que com este valor 90% das vezes é escolhida uma acção aleatória que não considera o valor da função acção-valor (equação 6) para o estado em que o robô se encontra.

Conforme esperado, também é possível constatar que o nº de iterações descresce com a diminuição da taxa de exploração.

A influência da taxa de aprendizagem (α) no desempenho do algoritmo não é visível já que o número de iterações se mantém semelhante ao variar este valor mantendo o factor de desconto e taxa de exploração constantes excepto com valores altos da taxa de exploração ($\epsilon = 0.9$).

Para valores baixos da taxa de exploração ($\epsilon = 0.3$) o nº de iterações mínimo é obtido com o factor de desconto médio ($\gamma = 0.5$), mas o aumento da taxa de exploração desfaz esta tendência.

4.3 Trabalho relacionado

O robô foi inspirado no trabalho de Vamplew [8] onde o robô é construído com base no Lego RCX e é utilizado o algoritmo SARSA [7]. O Lego RCX é a versão anterior ao Lego NXT Mindstorms, tendo este um processador mais potente e sensores fazem leituras mais fidedignas.

O algoritmo SARSA[7] é semelhante ao Q-learning e as suas diferenças são mínimas: o algoritmo usa a política para actualizar os valores de $Q(s, a)$, ou seja, os valores de $Q(s, a)$ são actualizados com base na política que está a ser seguida. O Q-learning actualiza os valores utilizando a política de examinação.

O robô construído também utiliza dois sensores de infravermelhos, postos lado a lado na parte frontal, sendo o objectivo medir o número de iterações que o robô necessita para aprender a seguir o percurso sem o perder.

Em [8], o robô demora em média 20 minutos a aprender a seguir a linha; neste trabalho, a experiência com o pior desempenho do robô demorou 8 minutos e na melhor configuração ($\alpha = 0.3$, $\gamma = 0.5$ e $\epsilon = 0.3$) o robô demora cerca de 2 minutos.

5 Conclusões

Este artigo estuda o algoritmo de aprendizagem por reforço, o Q-learning, e a sua interacção com o método de pesquisa ϵ-*Greedy* através do comportamento

[7] Acrómino inglês para *state-action-reward-state-action*.

apresentado por um robô cujo objectivo é seguir um percurso. Foi analisada a infuência e interacção dos diversos parâmetros existentes: a taxa de aprendizagem α, o factor de desconto γ (parâmetros do algoritmo) e a taxa de exploração ϵ (parâmetro da pesquisa).

Pelas experiências realizadas é possível concluir que o melhor desempenho do robô é obtido para a configuração $\{\alpha = .3, \gamma = .5, \epsilon = .3\}$ (taxa de aprendizagem e de exploração baixas e factor de desconto médio).

As experiências realizadas também confirmam que valores altos da taxa de exploração dão origem a piores desempenhos do robô, já que o seu comportamento torna-se errático.

Por outro lado, não foi possível verificar a influência da taxa de aprendizagem no comportamento do robô já que o nº de iterações necessárias para a sua aprendizagem é semelhante quando a taxa é alterada e se mantêm constantes os restantes parâmetros.

Como trabalho futuro pretende-se implementar o método de pesquisa *Softmax* e comparar o seu desempenho com o método de pesquisa ϵ-*Greedy* aqui apresentado.

Pretende-se também construir um robô com um objectivo diferente como seguir uma fonte de luz ou som. Ao contrário do robô aqui apresentado, naquele haverá um estado final que o robô deve atingir.

Referências

1. Figueira, O.R.G.: DROIDE M.L.P Potencializando a Plataforma. Master's thesis, Universidade da Madeira (2008)
2. Gomes, V.M.: Controle Inteligente de Tempo Livre em Tutoria Multissessão. Master's thesis, Universidade Federal de Goiás (2009)
3. Haykin, S.: Neural Networks: A Comprehensive Foundation Second Edition. Prentice Hall (1998)
4. Kaelbling, L., Littman, M., Moore, A.: Reinforcement learning: A survey. Journal of Artificial Intelligence Research 4, 237–285 (1996)
5. Minsky, M.: Steps toward artificial intelligence. In: Computers and Thought. pp. 406–450. McGraw-Hill (1961)
6. Moral, J.A.B.: Multithreading with Java leJOS (2008)
7. Sutton, R.S., Barto, A.G.: Reinforcement Learning: An Introduction. MIT Press (1998)
8. Vamplew, P.: Lego™ mindstorms™ robots as a platform for teaching reinforcement learning. In: International Conference on Artificial Intelligence in Science and Technology. pp. 21–25 (2004)
9. Watkins, C.J.C.H.: Learning from Delayed Rewards. Ph.D. thesis, Cambridge University, Cambridge, England (1989)
10. Wyatt, J.: Issues in putting reinforcement learning onto robots. In: Mobile Robotics Workshop, 10th Biennial Conference of the AISB. (1995)

Integração do Facebook com um veículo automóvel

Informação social como filtro de conteúdos de localização

Paulo Amaral

Universidade de Évora, Portugal
m5682@alunos.di.uevora.pt

Resumo Com o crescimento da utilização das redes sociais na Internet que se tem assistido nos últimos anos, a integração das mesmas em diversas aplicações informáticas tornou-se prática comum. Existem já casos onde foi feita a integração de redes sociais com veículos automóveis, utilizando as mesmas para diversos propósitos. Neste artigo é discutida a vantagem da utilização das informações obtidas numa rede social na filtragem de conteúdos a aplicar num sistema de marcação de pontos geográficos, graficamente representados através da visualização dos mesmos num mapa disponível num veículo. Ao enunciar os diversos aspectos de desenvolvimento de uma aplicação que utiliza informação social na marcação de dados geográficos num mapa, o artigo demonstra a utilidade do uso de contactos sociais de um indivíduo na filtragem de conteúdos relevantes durante o manuseio de um veículo.

1 Introdução

Podemos definir o termo rede social como "um conjunto de serviços *web-based* que permitem aos seus utilizadores (1) a construção de perfis públicos ou semi-públicos dentro de um sistema limitado, (2) a articulação de uma lista de outros utilizadores com quem partilham uma ligação, e (3) o acesso à lista de ligações de um utilizador, bem como às listas de ligações de outros utilizadores do sistema" [1]. Assistiu-se nos últimos anos a um enorme crescimento da utilização de redes sociais na Internet. As diversas redes sociais disponíveis (*Facebook*, *Twitter*, *MySpace*, etc.) são, hoje em dia, ferramentas do uso comum de várias pessoas e empresas de todo o mundo, assumindo-se por isso como as principais ferramentas de ligação e interação social à escala global [7]. A plataforma *Facebook*, em particular, é a plataforma de rede social mais utilizada na Internet, permitindo um número de ligações e partilha de conteúdos a um maior nível de pessoas que qualquer outra rede social [9].

O Gecko Merula é um veículo eléctrico produzido em Évora com o apoio de estudantes da Universidade de Évora [4]. É também para este veículo que o autor deste artigo está a realizar um trabalho, contextualizado nas dissertações de mestrado de engenharia informática da Universidade de Évora, que tem como objectivo o desenvolvimento de uma interface para o veículo. Esta interface deverá ser responsável pela visualização, por parte do condutor do veículo, dos

componentes habituais de um automóvel eléctrico (velocímetro, nível de bateria, etc.), sendo também esta interface responsável pela interação do condutor com o sistema de navegação por GPS e com o sistema de interação com redes sociais, sendo sobre este último ponto que o artigo se irá focar. Importa então perceber de que forma as ligações estabelecidas pelos utilizadores do *Facebook* podem dar, ao condutor, um contributo útil durante a condução de um veículo. Este artigo irá então debruçar-se, no contexto da aplicação das possibilidades das redes sociais nas mais diversas aplicações informáticas, na integração da plataforma Facebook nos serviços disponíveis num veículo automóvel. É importante, no entanto, referir que o objectivo do artigo não é o de ser um guia intensivo das tecnologias utilizadas na integração do *Facebook* com um veículo. O objectivo do artigo é sim o de discutir a relevância que uma rede social previamente desenvolvida e populada pode ter no decorrer de uma viagem de um veículo que se conecte a uma ou mais redes sociais. É também importante referir que este artigo se limita ao estudo da integração de redes sociais em veículos, e não à utilização de redes sociais para veículos (sem integração da rede social no mesmo). A utilização (sem integração) de redes sociais para veículos é aquela na qual a rede social pode ou não ser acedida, directa ou indirectamente, a partir do veículo e que não fornece informações que auxiliem a realização de alguma tarefa durante o acto da condução (por exemplo, uma rede social na qual os seus utilizadores trocam informações acerca de carros usados). Uma rede social integrada num veículo é aquela na qual a rede social é acedida, directa ou indirectamente, a partir do veículo (podendo também conter serviços específicos que são acedidos fora do veículo) e que fornece informação social que auxilie na realização de uma ou mais tarefas durante o acto da condução, e é precisamente acerca desta última forma de utilização de redes sociais que o artigo se refere.

2 Trabalho relacionado

A integração de redes sociais em veículos automóveis não é uma ideia que surgiu com a expansão mundial das redes sociais nos finais da década de 2000. Como exemplo desta situação, existe uma dissertação de mestrado datada de 2006 [5] na qual o autor propôs que se utilizassem as redes sociais mais populares da altura (*Orkut*, *Friendster*, etc.) de forma a desenvolver um sistema para um veículo em que se combinassem informações de localização e informações sociais. As informações de localização dos veículos eram dadas pelos sistemas GPS dos mesmos, enquanto que as informações sociais eram dadas pela ligação dos veículos a redes sociais. Segundo o autor da dissertação, o acesso ao grupo de amigos do condutor, bem como às informações de localização dos mesmos, permitia ao sistema de navegação por GPS do veículo uma filtragem do conteúdo demonstrado no mapa. Este conteúdo podia materializar-se em "sítios de interesse" marcados pelos utilizadores, ou mesmo na localização actual ou futura do veículo. A utilização da filtragem de conteúdos com auxílio a redes sociais significa que o sistema ganhou a capacidade de apresentar informação socialmente relevante ao condutor, i.e., ao limitar os conteúdos apresentados no mapa (apre-

sentando somente aqueles fornecidos pelos utilizadores considerados amigos do condutor na rede social), o condutor pode limitar a sua atenção a informações relevantes dentro do seu círculo social. Esta limitação de conteúdos ao nível social é um conceito fulcral para o presente artigo, e a sua aplicação prática será discutida mais abaixo.

Mais recentemente, uma equipa de alunos da Universidade de Michigan desenvolveu uma aplicação, denominada *Caravan Tracker*, que apresentou um conceito diferente do proposto na dissertação de mestrado referida no parágrafo anterior, visto que a aplicação *Caravan Tracker* não pretende a utilização de redes sociais já existentes, mas sim estabelecer ligações sociais entre um conjunto de viajantes. As ligações estabelecidas através da utilização da aplicação permitem aos seus utilizadores a partilha de informações acerca da quantidade de combustível existente nos seus veículos (e comparar essa informação), bem como a partilha de "pontos de interesse" entre os indivíduos deste círculo social [2].

Algumas marcas comerciais começam já também a integrar redes sociais nos seus veículos, como é o caso da *Toyota*, que se encontra a desenvolver, no presente momento de escrita deste artigo, uma rede social para os seus veículos eléctricos. Esta rede social pretende oferecer diversas funcionalidades ao seu utilizador, tal como o envio de um aviso, para o telemóvel do condutor, de que o nível de energia do seu veículo é baixo. A *Toyota* pretende ainda que o veículo possa comparar o seu estado actual com o estado de outros veículos idênticos, recorrendo-se dessa mesma ligação à rede social [8]. Outra marca comercial que já explorou as possibilidades da utilização das redes sociais em veículos foi a *Ford*, cujos engenheiros realizaram uma viagem pelos Estados Unidos na qual o veículo em que viajaram enviava mensagens para a rede social *Twitter*. As mensagens enviadas pelo veículo da *Ford* eram informações acerca do estado do próprio veículo ou de condições alheias ao veículo (o estado do tempo, o trânsito, etc.), sendo que as informações alheias ao veículo eram dadas por sensores ligados ao mesmo [10].

3 A aplicação VeículoSocial - Arquitectura e Tecnologias

Nesta secção é apresentada a aplicação **VeículoSocial**, que à data de escrita deste artigo se encontra em desenvolvimento, e que tem como propósito a definição de um sistema, para um veículo, que se liga à rede social *Facebook* para definir, durante o acto de condução da viatura, pontos geográficos relevantes ao condutor que poderão ser visualizados num mapa presente no ecrã do veículo.

3.1 Serviços disponíveis na aplicação

Os serviços que a aplicação disponibilizará ao condutor serão:

1. A disponibilização de "pontos de interesse" relevantes para o utilizador
2. A disponibilização de uma função que permita, a qualquer utilizador da aplicação, informar os seus amigos que este se encontra num determinado

ponto geográfico, por um determinado período de tempo. Este serviço funcionará de forma semelhante ao serviço *Google Latitude*, que é um serviço que permite aos seus utilizadores partilharem as suas localizações actuais com os seus contactos de serviços da *Google*, permitindo também este serviço a comunicação entre utilizadores através de *SMS*, *Gmail*, *Google Talk*, etc. [3]

Para concretização do serviço 1, a aplicação deve efectuar uma ligação a um servidor que contém todos os utilizadores registados no sistema. Para cada um dos utilizadores registados, o servidor deve também armazenar todos os pontos geográficos armazenados pelos mesmos, agrupando estes pontos em relação à sua proximidade a espaços que se incluem numa das seguintes categorias:

1. **Restauração**, onde se encontram todos os pontos relativos a estabelecimentos de restauração (restaurantes, cafeterias, etc.)
2. **Alojamento**, onde se encontram todos os pontos relativos a estabelecimentos que proporcionam alojamento (hóteis, pousadas, etc.)
3. **Postos de abastecimento**, onde se encontram todos os pontos relativos a postos de abastecimento de combustível (postos de abastecimento de gasolina ou gasóleo, postos de abastecimento eléctrico, etc.)
4. **Lazer**, onde se encontram todos os pontos relativos a áreas de lazer (cinemas, teatros, discotecas, etc.)
5. **Espaços Comerciais**, onde se encontram todos os pontos relativos a estabelecimentos destinados ao comércio (centros comerciais, supermercados, etc.)
6. **Parques de Estacionamento**, onde se encontram todos os pontos relativos a parques de estacionamento
7. **Outros**, onde se encontram todos os pontos que não se incluem numa das categorias anteriores

A interface da aplicação deve então mostrar num mapa os pontos identificados pelos utilizadores da aplicação, num determinado nível de proximidade à posição actual do veículo, pontos estes que deverão ser facilmente distinguidos em relação à sua categoria, e deve também a interface da aplicação permitir que o condutor possa filtrar, por categoria, os pontos apresentados no mapa. A aplicação permitirá, adicionalmente, que um utilizador confira uma avaliação (numa escala de 1 a 5) de um ponto previamente marcado, de modo a que os restantes utilizadores possuam uma informação mais completa dos espaços a que se dirige o seu veículo.

A aplicação permitirá ainda um outro tipo de filtragem do conteúdo a apresentar no mapa, a filtragem de conteúdos de acordo com as informações disponibilizadas pelos amigos do condutor. Para realização desta tarefa, a aplicação deve receber informação de uma rede social para filtragem de conteúdos relevantes ao condutor. A rede social escolhida é o *Facebook*, pois é a rede social mais utilizada e, portanto, existe uma maior probabilidade de o condutor possuir uma

rede de amigos já estabelecida nesta plataforma, sendo que a aplicação utilizará as ligações já estabelecidas pelos indivíduos no Facebook e aproveitar-se-á dessas mesmas ligações na filtragem de conteúdos. Esta filtragem resultará então de uma interpretação que o sistema realiza da informação social que recebe do *Facebook*: ao utilizador é dada a possibilidade de escolher entre a visualização, no mapa, dos pontos marcados por todos os utilizadores, ou entre os pontos marcados por utilizadores do sistema que são, simultaneamente, amigos do utilizador no *Facebook*. De igual forma, ao escolher este filtro, o condutor apenas terá acesso à avaliação dos pontos realizada pelos seus amigos do *Facebook*. Esta função irá ser adicionada ao sistema de forma a prevenir que os seus utilizadores utilizem informação de localização que não lhes é relevante, pois se o sistema não permitisse uma filtragem de conteúdos através das ligações sociais do utilizador, qualquer entidade pode adicionar pontos geográficos no mapa com um intuito contrário ao de fornecer informações de maior utilidade possível ao condutor. Por exemplo, imagine-se uma situação na qual o condutor pretende obter os estabelecimentos de alojamento que se encontram nas proximidades da posição actual do seu veículo: se não for realizada uma filtragem dos alojamentos de acordo com a experiência dos amigos do utilizador no *Facebook*, o condutor poderá escolher um estabelecimento de alojamento que tenha tido uma avaliação propositadamente exagerada por indivíduos a quem essa avaliação enganosa interessa. O filtro de conteúdos de localização confere ao utilizador, portanto, um significado socialmente relevante à função de visualização de "pontos de interesse" disponível na aplicação VeículoSocial.

Para concretização do serviço 2, a aplicação deve permitir que um utilizador envie, para um servidor, a sua posição actual bem como uma período de tempo estimado no qual o utilizador irá estar naquela determinada localização. A aplicação disponibilizará então esta informação de localização do indivíduo a todos os utilizadores do sistema que são amigos desse mesmo indivíduo no *Facebook*. Assim, um condutor poderá visualizar no mapa todos os seus amigos que estão num determinado raio de acção e que disponibilizaram a sua localização de forma a serem encontrados pelos indivíduos que lhes são socialmente relevantes. Tal como referido na descrição do serviço anterior, a informação social retirada do *Facebook*, ao funcionar novamente como uma filtragem de conteúdos de localização, apresenta-se fundamental para os utilizadores da aplicação, pois não interessará a um utilizador do sistema que, ao utilizar esta função, tenha a possibilidade de ser localizado por qualquer pessoa registada no sistema, mas sim somente por aqueles que mantêm uma ligação social com o utilizador que disponibiliza a sua localização actual. Este serviço, ao auxiliar o encontro físico dos seus utilizadores, adiciona uma maior dimensão social à aplicação VeículoSocial.

Refira-se também que a interface da aplicação deve ser construída de forma a que estes serviços sejam disponibilizados de uma forma intuitiva ao condutor, de modo a evitar ao máximo a distração do mesmo à tarefa primária da condução. De forma a que o condutor possa direcionar a maior parte da sua atenção à estrada, a interface definida do sistema deve fornecer uma interação simples e deve também tentar garantir que os seus utilizadores obtenham uma

rápida familiarização com o modelo de interação condutor-aplicação disponibilizado (por exemplo, através da utilização de ícones gráficos facilmente visíveis e reconhecíveis pelo condutor).

3.2 Arquitectura e Tecnologias

A arquitectura definida para o desenvolvimento da aplicação VeículoSocial pode ser consultada na figura 1. Os elementos constituintes do sistema que compõe a aplicação são: a Interface, o Veículo, o Backend do sistema, uma base de dados (pertencente ao Backend) e a rede social *Facebook*. De forma a requisitar um serviço à aplicação, o utilizador acede (A) à interface do sistema, que por sua vez irá requisitar, ao sistema instalado no veículo, as funcionalidades desejadas pelo utilizador (B). As funcionalidades requeridas pelo utilizador necessitam de dados do Backend e do Facebook e, por isso, o Veículo necessita de realizar pedidos ao Backend do sistema, que não é mais que uma aplicação que está instalada num servidor e que contém uma base de dados. A autenticação do utilizador, durante o acesso do Veículo ao Backend, deverá ser feita com o nome de *login* que o utilizador usa para aceder ao Facebook de forma a facilitar o cruzamento de dados no Backend. A password de autenticação do utilizador no sistema não deve, no entanto, ser a mesma que este utiliza para se autenticar no *Facebook*, por questões de privacidade. Assim, é necessária uma ligação entre o Veículo e o *Facebook* (F), ligação essa que retornará ao Veículo uma *token* temporária de acesso ao *Facebook* que será passada posteriormente ao Backend. O utilizador poderá necessitar de dados relativos a todos os utilizadores do sistema acerca de pontos geográficos que estejam num raio de x quilómetros, pelo que neste caso o Veículo apenas necessitá de realizar o pedido ao Backend (C) que, por sua vez, irá consultar a base de dados (E) e devolverá ao Veículo (C) os dados, relativos a pontos num raio de x quilómetros do veículo, de todos os utilizadores do sistema. Por outro lado, o utilizador poderá necessitar de dados de pontos geográficos, que estejam num raio de x quilómetros do veículo, relativos a utilizadores do sistema que são seus amigos no *Facebook*. Neste caso, o Veículo efectuará esse pedido ao Backend (C), pedindo este os dados relevantes à base de dados (E), dados esses que serão cruzados com as lista de amigos do utilizador no *Facebook*, que será obtida através de uma ligação do Backend com o *Facebook* (D). Uma vez obtidos e cruzados os dados necessários, o Backend devolve ao Veículo (C) os dados relativos a pontos num raio de x quilómetros assinalados pelos amigos do utilizador. Assim que o Veículo obtenha os pontos geográficos necessários que tem de assinalar no mapa, estes passarão a estar disponíveis à Interface (B), que permitirá a visualização dos mesmos ao utilizador (A).

Encontrando-se a aplicação VeículoSocial numa fase de desenvolvimento, não se podem listar com precisão todas as tecnologias que estão envolvidas na implementação do sistema. Assim, irá ser apresentada seguidamente uma breve lista de tecnologias previstas, no momento de escrita do artigo, a utilizar no desenvolvimento da aplicação. A Interface e o sistema presente no Veículo poderão ser desenvolvidos na plataforma *Android*, permitindo assim uma correcta implementação de funcionalidades num dispositivo móvel e de acesso por *touch screen*.

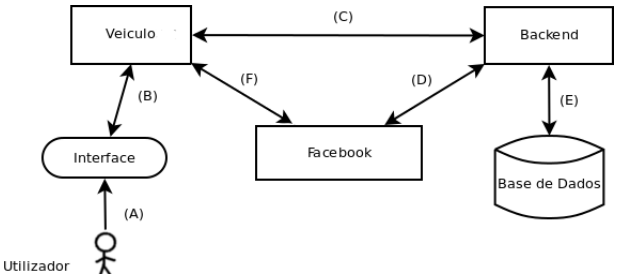


Figura 1. Arquitectura da aplicação VeículoSocial

O Backend, por sua vez, poderá ser definido como um *Lamp Server*, permitindo assim que o Veículo efectue uma conexão ao Backend através de uma camada de segurança SSL. De forma a comunicar com o *Facebook*, o Backend utilizará a API do Facebook que permite obter a lista de amigos de um utilizador através de um *token* de autenticação criado com os dados desse mesmo utilizador. Finalmente, o mapa a apresentar na aplicação deverá ser obtido através de uma ligação ao *Google Maps*, o que permite uma constante actualização do espaço físico representado no mapa, bem como uma simples forma de representar pontos geográficos no mesmo.

4 Conclusão e Trabalho Futuro

A utilização de informação social, proveniente das ligações sociais de um indivíduo numa rede social, na filtragem de conteúdos de localização pode ser útil para a obtenção de uma lista de pontos geográficos relevantes para o utilizador de um sistema que forneça informações de localização nas mais diversas categorias. A aplicação VeículoSocial, descrita neste artigo, permite a limitação da apresentação de dados geográficos referentes a localizações assinaladas no sistema por amigos do utilizador na rede social *Facebook*. A limitação de dados ao nível do círculo social não é uma ideia nova, tendo sido apresentada, por exemplo, na dissertação [5]. No entanto, ao contrário da dissertação referida anteriormente, a aplicação VeículoSocial faz uso da rede social mais popular da actualidade, o *Facebook*, pelo que existe uma maior probabilidade de o número de ligações sociais estabelecidas entre os seus utilizadores ser maior. Por outro lado, a aplicação VeículoSocial apresenta também um sistema de avaliação de conteúdos, disponível para qualquer um dos seus utilizadores, que permite uma maior filtragem de conteúdos relevantes a apresentar ao condutor.

Como a aplicação VeículoSocial não está ainda completamente implementada, será necessário perceber como serão apresentados os dados de localização ao condutor, e como será feita a interação do condutor com o sistema de modo a evitar a distração do mesmo no manuseio do seu veículo. Tendo em vista a disponibilização das funcionalidades da aplicação descritas no artigo, pode ser consultado um primeiro esboço da interface do sistema, para o serviço de

marcação de pontos de interesse no mapa, na figura 2. Neste esboço, podemos verificar que os botões de interação com o utilizador ocupam uma área significativa do ecrã e, de forma a promover uma rápida apreensão do funcionamento da interface por parte do utilizador, são utilizados ícones gráficos nos botões de seleção de limitação de conteúdos por todos os utilizadores do sistema ou por amigos do *Facebook*, tal como são utilizados ícones gráficos nos botões de filtragem de categorias para marcação de pontos geográficos no mapa, de acordo com as recomendações da *Alliance of Automobile Manufacturers*, entidade composta por vários fabricante de carros (*BMW Group*, *Ford Motor Company*, *General Motors Company*, etc.) [6].

Figura 2. Primeiro esboço da interface da aplicação

Em relação ao trabalho futuro a realizar para este projecto, deve ser, obviamente, realizada a implementação do sistema, de acordo com a arquitectura definida neste artigo. Esta mesma arquitectura poderá, no entanto, ser revista de modo a permitir que sejam efetuadas ligações a diversas redes sociais, não limitando, assim, o conteúdo social da aplicação à plataforma *Facebook*. Poderá ainda ser estudada a possibilidade de adicionar um novo filtro aos conteúdos a adicionar no mapa, pois de acordo com as funcionalidades definidas anteriormente, não é possível para um utilizador do sistema a limitação de conteúdos dentro do seu círculo de amigos. A adição desta funcionalidade deve ser estudada com precaução, pois a escolha, por parte do condutor, de um grupo de utilizadores da sua lista de amigos, enquanto conduz o veículo, é uma tarefa que se prevê demasiado complexa e distractiva. Uma solução para este problema pode ser o desenvolvimento de uma funcionalidade, que será acedida fora do veículo, na qual o utilizador pode previamente "construir" um ou mais grupos de amigos que o mesmo considera que apresentarão dados relevantes para si, podendo o utilizador filtrar a visualização de dados no mapa de acordo com estes grupos de amigos durante a utilização da aplicação VeículoSocial no seu veículo.

Referências

1. Danah M. Boyd Ellison and Nicole B. Ellison. Social network sites: Definition, history, and scholarship. *Journal of Computer-Mediated Communication*, 2007.

2. GigaOm. Social networked cars: The future of connected vehicles? http://gigaom.com/cleantech/social-networked-cars-the-future-of-connected-vehicles/, May 2010.
3. Google. See where your friends are with google latitude. http://googleblog.blogspot.com/2009/02/see-where-your-friends-are-with-google.html, April 2009.
4. Exame Informática. Já há quem fabrique carros elétricos em Évora. http://aeiou.exameinformatica.pt/ja-ha-quem-fabrique-carros-eletricos-em-evora=f1007259, September 2010.
5. Philip Angus Liang. Social networking in vehicles. Master's thesis, Massachusetts lnstitute of Technology, 2006.
6. Alliance of Automobile Manufacturers. Statement of principles, criteria and verification procedures on driver interactions with advanced in-vehicle information and communication systems. Technical report, Alliance of Automobile Manufacturers, 2006.
7. Arab Social Media Report. Facebook usage: Factors and analysis, January 2011.
8. Consumer Reports. Toyota to create a social network for its cars and drivers. http://news.consumerreports.org/cars/2011/05/toyota-to-create-a-social-network-for-its-cars-and-drivers.html, May 2011.
9. Internet World Stats. Facebook users in the world. http://www.internetworldstats.com/facebook.htm, June 2011.
10. The New York Times. Social networking for cars. http://wheels.blogs.nytimes.com/2010/07/20/social-networking-for-cars/, July 2010.

Índice